TIETO- JA SÄHKÖTEKNIIKAN TIEDEKUNTA

Ville Rompasaari
Kalle Palokangas

PUHEENEROTTELUJÄRJESTELMÄN TOTEUTUS INMOOV-ROBOTILLE

Kandidaatintyö
Tietotekniikan tutkinto-ohjelma
Toukokuu 2020

Rompasaari V., Palokangas K. (2020) Puheenerottelujärjestelmän toteutus InMoov-robotille. Oulun yliopisto, Tietotekniikan tutkinto-ohjelma, 44 s.

TIIVISTELMÄ

Koneellisia kuulojärjestelmiä ja niiden osia on kehitetty jo vuosikymmeniä; olemassa on kuitenkin edelleen useita ongelmia, jotka ovat esteenä ihmistä vastaavan kuulojärjestelmän saavuttamisessa. Yksi näistä ongelmista on puhujien erottelu puhesekoituksesta erillisiksi äänisignaaleiksi, jota kutsutaan myös cocktailkutsuongelmaksi. Vaikka ihmisen on helppo paikantaa ja erotella eri puhujat usean samanaikaisen puhujan joukosta, samaan suorituskykyyn yltävä koneellinen toteutus on osoittautunut haastavaksi. Usein ratkaisuissa pyritään hyödyntämään useista mikrofoneista koostuvia mikrofoniryhmiä, jotka mahdollistavat monikanavaisten kaiun- ja kohinanpoistomenetelmien sekä äänilähteiden suuntien käytön apuna erotteluprosessissa. Viime vuosina on myös tutkittu syväoppimista hyödyntäviä menetelmiä, jotka ovat antaneet lupaavia tuloksia.

Tässä työssä esitellään uPIT-syväoppimismenetelmää käyttävä toteutus puheenerottelujärjestelmästä ROS-ympäristössä InMoov-robotille. Työn tavoitteena on selvittää erottelualgoritmin tuoma hyöty robotin kuulojärjestelmän osana. Toteutettu ROS-komponentti antaa muille järjestelmän komponenteille rajapinnan, joka tarjoaa robotin ympärillä kuuluvien puhujien erotellut puhesignaalit, ja lisäksi estimaatin yhden puhujan suunnan atsimuutista astelukuna suhteessa pään katsesuuntaan. Ratkaisussa on käytetty Seeed Studion ReSpeaker Mic Array v2.0 -mikrofonijärjestelmää, joka suorittaa sisäänrakennetusti kaiun ja taustamelun vaimennuksen, keilanmuodostuksen ja äänen tulosuunnan estimoinnin. Mikrofonin tallentama puhdistettu signaali välitetään uPIT-syväoppimismenetelmän avulla koulutettuun puheenerottelualgoritmiin, joka erottelee eri puhujille kuuluvat signaalit toisistaan.

Erottelun tuloksena testiaineistolla saavutettiin parhaimmillaan 5,99 dB parannus signaali-särösuhteessa kahden vastakkaista sukupuolta olevan aiemmin nähdyn puhujan erottelussa. Uusien puhujien erottelussa vastaava arvo on 5,60 dB. Koska tulokset saatiin käyttäen LibriSpeech-kieliaineistoa yleisen puheenerotteluun käytetyn WSJ0-aineiston sijasta, arvot eivät ole täysin vertailukelpoisia vastaavanlaisten tutkimusten kanssa. Vaikka saadut tulokset ovat parempia kuin joillain tavanomaisilla yksikanavaisilla puheenerottelumenetelmillä saavutetut arvot, kehitetyn puheenerottelujärjestelmän ei nähdä yltävän käytännön tilanteiden vaatimaan suorituskykyyn. Järjestelmä tarjoaa kuitenkin hyvän lähtökohdan robotin puheenerottelulle.

Avainsanat: puheenerottelu, uPIT, konekuulo, robotiikka

Rompasaari V., Palokangas K. (2020) Speech Separation System Solution for InMoov Robot. University of Oulu, Degree Programme in Computer Science and Engineering, 44 p.

ABSTRACT

Machine hearing systems and their subcomponents have been researched for decades; however, there are still problems that are preventing the system from reaching human-like performance. One of the problems is separating multiple speakers from a speech mixture into separate signals, which is called the cocktail party problem. Even though it is easy for humans to locate and separate different speakers from a group of multiple simultaneous speakers, achieving this kind of performance in a machine has proven to be a challenging task. Often the proposed solutions use an array of multiple microphones, which open up the possibility of using multichannel dereverberation and noise suppression techniques and directions of sound sources to aid the separating process. Recent years have also seen increasing research of solutions using deep learning, which have given promising results.

In this thesis, a speech separating system using the uPIT deep learning technique for InMoov humanoid robot is presented. The goal of the thesis is to see whether the speech separating system brings any meaningful improvements to the machine hearing system in the system's ability to process speech. The developed component provides the other components of the system an interface for accessing separated speech signals and an estimate of the azimuth direction of one of the speakers. The solution utilizes Seeed Studio's ReSpeaker Mic Array v2.0 microphone array, which provides built-in functionality for dereverberation and noise suppression, beamforming, and estimation of the direction of sound sources. The recorded and processed sound signals are sent to a deep learning speech separation system trained with utterance level permutation invariant training, which separates the different speech signals.

The separation system achieved at most a 5.99 dB improvement in signal-to-distortion ratio with two speakers of different genders in closed condition. In open condition, the improvement was 5.60 dB. Because the results were acquired using the LibriSpeech dataset, instead of the more common WSJ0 dataset, as the training data of the model, the results are not comparable to other similar studies. Even though the SDR values show improvement in results over some of the other single-channel separation methods, the performance of the system was not deemed good enough to meet the requirements of real-world applications. However, the system is still a good starting point for further development of the robot's hearing system.

Keywords: speech separation, uPIT, machine hearing, robotics

SISÄLLYSLUETTELO

ALKULAUSE

Työ on laadittu osana Oulun yliopiston tietotekniikan alan kurssia 521275A, Sulautettujen ohjelmistojen projekti. Tahdomme kiittää Teemu Tokolaa kurssin ja työn ohjaamisesta, ja arvokkaista neuvoista tämän ja tulevien töiden tekemiseen, sekä prof. Juha Röningiä työn tarkastamisesta. Kiitämme myös Juho Kokemäkeä osallistumisesta työn suunnitteluvaiheeseen.

Oulussa 27. toukokuuta 2020

Ville Rompasaari
Kalle Palokangas

LYHENTEIDEN JA MERKKIEN SELITYKSET

AED	Adaptiivinen ominaisarvohajotelma, Adaptive Eigenvalue Decomposition
ASA	Kuulema-analyysi, Auditory Scene Analysis
ASR	Automaattinen puheentunnistus, Automatic Speech Recognition
BSS	Sokea lähteiden erottelu, Blind Source Separation
CASA	Laskennallinen kuulema-analyysi, Computational Auditory Scene Analysis
CC	Nähtyjen puhujien joukko, Closed Condition
DOA	Äänen tulosuunta, Direction Of Arrival
DSB	Viivesummaus keilanmuodostin, Delay-and-Sum Beamformer
GCC-PHAT	Normalisoitu ristikorrelaatio vaihemuunnoksella, Generalized Cross-Correlation with PHAse Transform
HRTF	Pään siirtofunktio, Head Related Transfer Function
ICA	Riippumaton komponenttianalyysi, Independent Component Analysis
ILD	Korvien välinen tasoero, Interaural Level Difference
ITD	Korvien välinen aikaero, Interaural Time Difference
LSTM	Pitkä lyhytkestomuisti, Long Short-Term Memory
MSE	Keskineliövirhe, Mean Squared Error
MUSIC	Monisignaaliluokittelu, MUltiple SIgnal Classification
MVDR	Varianssin minimoiva särötön vaste, Minimum Variance Distortionless Response
NMF	Ei-negatiivisien matriisien tekijöihin jako, Non-negative Matrix Factorization
OC	Uusien puhujien joukko, Open Condition
PIT	Permutaatioinvariantti koulutus, Permutation Invariant Training
PSM	Vaiheherkkä maski, Phase Sensitive Mask
ReLU	Tasasuunnattu lineaariyksikkö, Rectified Linear Unit
RNN	Takaisinkytkeytyvä neuroverkko, Recurrent Neural Network
ROS	Robottikäyttöjärjestelmä, Robot Operating System
SDR	Signaali-särösuhde, Signal-to-Distortion Ratio
SRP-PHAT	Ohjattu vasteteho vaihemuunnoksella, Steered Response Power with PHAse Transform
TDOA	Tuloaikaero, Time Difference Of Arrival
uPIT	Lausahdustason permutaatioinvartiantti koulutus, utterance-level Permutation Invariant Training
VAD	Äänen aktiivisuuden tunnistus, Voice Activity Detection

1. JOHDANTO

Ihmisten kehittämä kieli mahdollistaa suuren informaatiomäärän pakkaamisen muutamaan sanaan tai lauseeseen, jotka voidaan nopeasti jakaa puheen kautta. Puheen kuulevat ja tiedostavat ihmiset tunnistavat puheen lähteen, käsittelevät puheen sisällön ja muodostavat siitä oman tulkintansa. Prosessi on pohjimmiltaan monimutkainen mutta ihminen suoriutuu siitä vaivatta. Koska puhe on ihmiselle niin luonnollinen tapa välittää tietoa, sitä pyritään hyödyntämään myös koneiden kanssa kommunikointiin.

Koneellisten kuulojärjestelmien, eli konekuulon, käyttö on lisääntynyt huomattavasti viime vuosina tietokoneiden suorituskyvyn kasvaessa. Nykyiset järjestelmät vaativat kohtuullisen määrän laskentatehoa, jota ei ole aiempina vuosina ollut saatavissa. Erityisesti tekoälyn suosion kasvu on edistänyt kuulojärjestelmien toimintaa esimerkiksi parantamalla algoritmien suorituskykyä meluisissa ja kaikuvissa ympäristöissä. Myös useista mikrofoneista koostuvien mikrofonijärjestelmien hintojen lasku on auttanut kuulojärjestelmien suosiota tuoden ulottuville suorituskyvyltään tehokkaampaa laitteistoa.

Nykyisillä kuulojärjestelmillä on mahdollista luotettavasti tunnistaa selkeästi lausuttua puhetta kevyen taustamelun seasta [1]. Ongelmia kuitenkin ilmenee esimerkiksi tilanteissa, joissa puhe on monimutkaista tai epäselvää, samanaikaisia puhujia on useita, tai taustamelua ja kaikua on liikaa. Suorituskyvyn parantaminen erityisesti epäselvän puheen kannalta on tärkeää mahdollisimman suuren käyttäjäkunnan tavoittamiseksi. Konekuulolla ei siis voida vielä täysin korvata perinteisiä koneiden kanssa käytettäviä kommunikaatiomenetelmiä.

Konekuulon eräitä sovelluskohteita ovat esimerkiksi vammaisten apuvälineet [2, 3], kuulolaitteet [4, 5] ja autot [6], jossa äänikomennoilla voidaan ohjata esimerkiksi ovien aukeamista tai suuntamerkkejä. Äänen avulla voidaan myös määrittää äänilähteiden sijainteja, josta on hyötyä esimerkiksi kameroiden suuntien ohjaamisessa [7]. Arjessa konekuulo tulee kuitenkin selvimmin esille viime vuosina suosioon nousseiden virtuaalisten avustajien, kuten Google Assistantin, Applen Sirin tai Amazonin Alexan kautta. Avustajat ovat käytössä esimerkiksi älypuhelimissa ja älykaiuttimissa ja mahdollistavat niihin yhdistettyjen laitteiden käyttämisen äänikomentoja hyödyntäen. Komennoilla voidaan esimerkiksi ohjata IoT-laitteita (*esineiden internet, Internet of Things*), kuten valoja, ovien lukkoja tai kaiuttimia.

Koska avustajia sisältävät laitteet ovat käytännössä jatkuvasti päällä olevia mikrofoneja, esille nousee kysymys yksityisyydensuojasta: Kuka takaa sen, että laitteet eivät analysoi ääntä jatkuvasti ja lähetä tuloksia esimerkiksi markkinoinnin tarpeisiin? Huomioon otettavaa on myös se, missä laitteet käsittelevät tallentamaansa ääntä. Yksityisyyden näkökulmasta tietojen paikallinen käsittely on tavoiteltavaa; useat tämän hetkiset toteutukset kuitenkin hyödyntävät pilvilaskentaa äänisignaalien prosessoinnissa, jonka vuoksi signaalien sisältämä informaatio täytyy lähettää laitteen ulkopuolelle [8].

Konekuulojärjestelmien kehityksen tavoitteena on saavuttaa järjestelmä, joka kykenee analysoimaan ääniä vähintään yhtä tehokkaasti kuin ihminen. Ihmisen tasolle yltävän koneellisen kuulojärjestelmän kehittäminen edellyttää kuitenkin useiden erillisten ongelmien selvittämistä, joista monet ovat yhä vailla ratkaisua. Ihmistä vastaavasti kuulevan koneen voisi olettaa kykenevän erottamaan puhe musiikista ja taustamelusta, tunnistamaan äänilähteiden suunnat, oppimaan tärkeät äänet ja

niiden merkitykset, ja reagoimaan ääniin reaaliajassa [9]. Tällaista kokonaisvaltaista koneellisesti toteutettua kuulojärjestelmää ei ole kuitenkaan vielä täysin saavutettu.

Yksi suurimmista haasteista on ympäristön aiheuttamien häiriötekijöiden huomioon ottaminen järjestelmän toiminnassa. Käytännön sovelluskohteissa esintyy aina jonkin verran melua, kuten liikenteen ääniä, puheensorinaa tai kaikua, jotka häiritsevät esimerkiksi puheenerottelua ja äänilähteiden sijaintejan määrittäviä algoritmeja. Mikäli häiriöitä ei käsitellä, niiden vaikutukset siirtyvät järjestelmässä eteenpäin heikentäen esimerkiksi puheentunnistuksen luotettavuutta.

Käytännön tilanteissa myös samanaikaisia puhujia voi olla useampi kuin yksi. Ongelmaksi muodostuukin, miten yhtäaikaiset puhujat saadaan eroteltua melusta, kaiusta ja muista puhujista, jotta kaikkien eri puhujien puheet voidaan käsitellä. Tätä ongelmaa kutsutaan cocktailkutsuongelmaksi [10]. Nimitys tulee ihmisen kyvystä kuulla ja keskittyä yhteen puhujaan cocktailkutsujen tapaisessa musiikin ja hälinän täyteisessä ympäristössä.

Tässä työssä esitellään InMoov-humanoidirobotille kehitetty toteutus eräästä cocktailkutsuongelman ratkaisuun käytetystä puheenerottelujärjestelmästä. Järjestelmä erottelee kahden samanaikaisen puhujan puheet toisistaan uPIT-syväoppimismenetelmää hyödyntäen erillisiksi puhesignaaleiksi, joita voidaan käsitellä itsenäisesti. Toteutuksessa käytettävä mikrofoni toteuttaa myös melun- ja kaiunpoiston äänen tallennusvaiheessa ja laskee estimaatin voimakkaimman puhujan suunnalle.

2. KONEKUULON HAASTEET

Toimivan konekuulojärjestelmän toteuttaminen edellyttää monenlaisten haasteiden ratkaisemista. Mikäli halutaan saavuttaa ihmisen tasolle yltävä suorituskyky, järjestelmän tulisi kyetä nopeasti erottelemaan eri äänilähteet toisistaan, paikantamaan äänilähteiden sijainnit ja käsittelemään äänisignaalien kantama informaatio. Usein kaikkien näiden ominaisuuksien toteuttaminen ei ole kuitenkaan tarpeellista, esimerkiksi kameroiden suuntia säätävälle järjestelmälle riittää pelkkä äänilähteiden suuntien estimointi, äänikomentoja kuuntelevalle järjestelmälle puolestaan luotettava puheentunnistus.

Lähes kaikissa käyttötapauksissa, keskeinen haaste on melun ja kaiun vaikutusten minimointi. Käytännön sovelluskohteissa ympäristö on harvoin hiljainen, jolloin erilaiset melun ja kaiun aiheuttamat häiriöt tulee ottaa huomioon järjestelmän toiminnassa. Yleensä tämä tapahtuu käyttäen jonkinlaista kaiun- ja kohinanpoistomenetelmää joko äänen tallennuksen yhteydessä, tai myöhemmin signaaleja käsiteltäessä. Jotkin nykyiset syväoppimismenetelmät kykenevät myös joissain tapauksissa oppimaan melun ja kaiun vaikutukset ja suodattamaan ne pois.

Useita mikrofoneja käytettäessä, yksi suosituimmista melun- ja kaiunpoistomenetelmistä on keilanmuodostus. Keilanmuodostus toimii painottamalla tietyistä suunnista saapuvia signaaleja ja vaimentamalla ympäröiviä ääniä. Näin saadaan muodostettua suunnattava keila, jonka avulla kuuntelu voidaan painottaa haluttujen äänilähteiden suuntaan. Keilan suuntaamiseksi tarvitaan tietoa äänien tulosuunnista.

Monille kuulojärjestelmille on hyödyllistä tietää mistä suunnista ympäröivät äänet tulevat. Tietoa suunnista voidaan hyödyntää keilanmuodostuksen lisäksi esimerkiksi videoneuvotteluiden kameroiden suuntaamisessa tai parantamaan puheenerottelun, sekä melun- ja kaiunpoistomenetelmien tehokkuutta. Suuntien tehokkaaseen estimointiin vaaditaan useasta mikrofonista koostuva mikrofoniryhmä; myös yhden mikrofonin toteutuksia on olemassa [11, 12] mutta niiden suorituskyky ei yllä useita mikrofoneja käyttävien menetelmien tasolle. Yleisimmät suuntienestimointimenetelmät käyttävät hyväksi eri mikrofonien tallentamien äänisekoitusten eroavaisuuksia, kuten äänisignaalien tuloaikaeroa.

Tilanteissa, joissa on tarpeen käsitellä usean samanaikaisen puhujan tuottamaa puhetta tarvitaan puheenerottelua. Puheenerottelumenetelmien tavoitteena on erotella eri puhujat useiden samanaikaisten puhujien ja taustamelun seasta erillisiksi puhesignaaleiksi, joita voidaan käsitellä itsenäisesti. Tätä ongelmaa kutsutaan myös cocktailkutsuongelmaksi, jolla viitataan ihmisen kykyyn kuulla ja keskittyä yhteen puhujaan coctailkutsujen tapaisessa meluisassa ympäristössä. Puheenerottelu voidaan toteuttaa joko yksikanavaisesti käyttäen vain yhtä äänisekoitusta tai monikanavaisesti hyödyntäen useilla mikrofoneilla tallennettuja äänisekoituksia.

Yksikanavaisia erottelumenetelmiä ovat esimerkiksi ei-negatiivisien matriisien tekijöihin jako (*Non-negative Matrix Factorization, NMF*) ja ihmisen kuulosta vaikutteita ottava laskennallinen kuulema-analyysi (*Computational Auditory Scene Analysis, CASA*). Myös viime vuosikymmenen aikana esiin nousseet syväoppimista hyödyntävät menetelmät, kuten syväklusterointi ja permutaatioinvariantti koulutus, ovat yksikanavaisia. Monien syväoppimismenetelmien on osoitettu saavuttavan

huomattavia parannuksia signaali-särösuhteessa perinteisiin CASA- ja NMF-menetelmiin verrattuna [13].

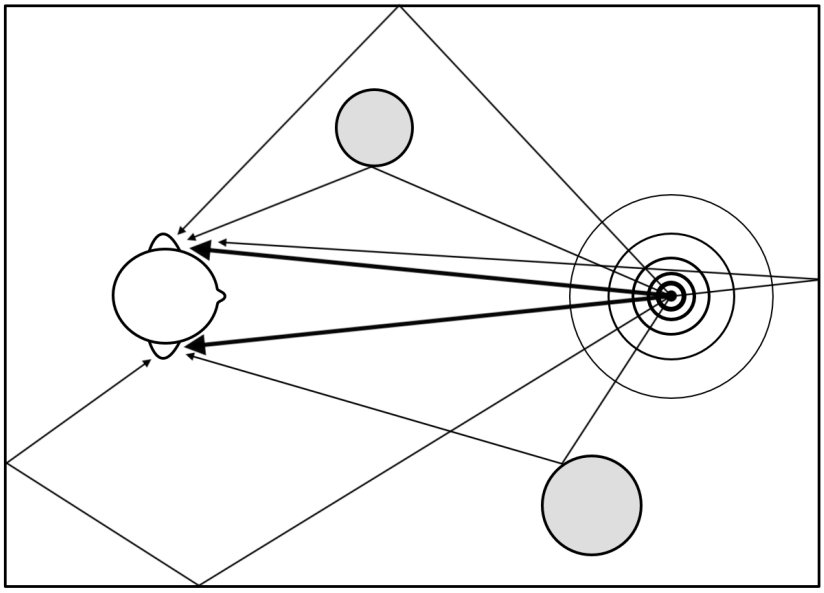
Yleisimmät monikanavaiset erottelumenetelmät hyödyntävät toteutuksissaan keilanmuodostusta tai sokean lähteiden erottelun menetelmiä. Keilanmuodostusmenetelmät käyttävät keilanmuodostusta vähentämään melun ja kaiun vaikutuksia äänisignaalissa ja tämän jälkeen erottelevat puheen jotain yksikanavaista menetelmää käyttäen. Sokean lähteiden erottelun menetelmät vertailevat eri mikrofonien tuottamia äänisekoituksia ja käyttävät signaalien tilastollisia piirteitä erottelun toteuttamiseksi.

Melun- ja kaiunpoistomenetelmien ja puheenerottelun yksi keskeisimmistä tavoitteista on muodostaa mahdollisimman selkeä puhesignaali puheentunnistuksen käsiteltäväksi. Puheentunnistuksen tarkoituksena on analysoida syötetty puhesignaali ja tunnistaa sen sisältämä informaatio. Suurin haaste puheentunnistuksessa häiriöpitoisten puhesignaalien käsittelyn lisäksi, on erilaisten puhujien huomioonottaminen, esimerkiksi puhujan kielitaito ja äänen rakenne vaikuttavat usein puheentunnistusalgoritmien luotettavuuteen.

Tietoturvan ja yksityisyyden näkökulmasta konekuuloa voidaan pitää jossain määrin ongelmallisena: vaikka järjestelmät ovat hyödyllisiä ja helpottavat koneiden kanssa kommunikointia, niiden toiminta kuitenkin usein edellyttää jatkuvasti päällä olevia mikrofoneja. Tämä herättää kysymyksen siitä, tallentavatko ja analysoivatko laitteet ääntä aina päällä ollessaan. Huomioon otettavaa on myös se, missä konekuulojärjestelmät käsittelevät tallentamaansa ääntä. Melun- ja kaiunpoisto sekä puheenerottelu suoritetaan usein paikallisesti laitteen sisällä; useat puheentunnistusjärjestelmät kuitenkin hyödyntävät pilvilaskentaa toteutuksessaan [8] eli tallennettu ääni täytyy lähettää laitteen ulkopuolelle. Yksityisyydensuojan ja tietoturvan osalta on tavoiteltavaa, että kaikki järjestelmän osat käsittelisivät dataa paikallisesti.

2.1. Melun ja kaiun poisto

Pistemäisen äänilähteen tuottaessa ääntä, kuulijan korvaan saapuu sekä suoraa että epäsuoraa ääntä kuvan 1 mukaisesti [14]. Suora ääni on ääntä, joka kulkee suoraan esteettä äänilähteestä kuulijaan. Epäsuoralla äänellä puolestaan tarkoitetaan kaikua, eli äänilähteen tuottaman äänen heijastumia ympäristöstä, jotka kuulija kuulee suoran äänen lisäksi [15]. Kaiussa on aina viivettä ja vaimennusta suoraan ääneen verrattuna johtuen pidemmästä matkasta kuulijaan ja akustisesta absorptiosta [16].



Kuva 1. Äänen kulkeutuminen suljetussa tilassa. Tummat nuolet kuvaavat kuulijan korvaan saapuvaa suoraa ääntä, ohuet nuolet epäsuoraa kaikua.

Kaiku voi olla toivottu ilmiö, esimerkiksi konserttisaleissa, jotka suunnitellaan tuottamaan mahdollisimman miellyttävä akustinen ympäristö soitettavalle musiikille [15]. Sen avulla voidaan myös tehdä päätelmiä ympäristöstä ja äänilähteen sijainnista ympäristön suhteen [14]. Liiallinen kaiku, ympäristön melun ohella, on kuitenkin konekuulon kannalta haitallista: äänen tulosuunta sumentuu ja ylimääräinen häly sotkee kiinnostuksen kohteena olevia suoria ääniä näin tehden puheen erottelusta ja tunnistuksesta haastavaa [15, 16, 17]. Negatiiviset vaikutukset äänisignaalin laatuun lisääntyvät äänilähteen ja kuulijan etäisyyden kasvaessa, ja mikäli kaiun ja melun intensiteetti nousee tarpeeksi suureksi, voivat ne peittää halutut äänisignaalit täysin [16]. Tämän vuoksi onkin kehitetty useita menetelmiä melun ja kaiun vaikutusten minimoimiseksi.

Häiriöpitoinen signaali $x(t)$ tietyllä ajanhetkellä t voidaan esittää puhtaiden äänisignaalien $s(t)$, kaiun $r(t)$ ja melun $n(t)$ summana [15]:

$$x(t) = s(t) + r(t) + n(t) \tag{1}$$

Melun- ja kaiunpoistomenetelmien tavoitteena on erottaa ja estimoida puhtaita äänisignaaleja mahdollisimman tarkasti häiriöpitoisesta signaalista minimoiden melun ja kaiun vaikutukset [16].

Sopivan menetelmän valinta riippuu siitä millaisia oletuksia tehdään olosuhteista, joissa ääni on tallennettu, esimerkiksi mikrofonien määrästä sekä ympäristön ja äänien ominaisuuksista [15]. Melun- ja kaiunpoistolle on sekä yksi- että monikanavaisia toteutuksia; monikanavaiset menetelmät ovat kuitenkin usein tehokkaampia, sillä ne voivat käyttää äänien tulosuuntia toteutuksessaan [18, 15]. Eräs tällaista avaruudellista suodatusta hyödyntävä menetelmä on keilanmuodostus.

Myös syväoppimiseen perustuvia menetelmiä on kehitetty [19]. Menetelmät toteuttavat vaimennuksen kouluttamalla neuroverkot tunnistamaan millaiset häiriöpitoiset signaalit vastaavat tietynlaisia puhtaita signaaleja [20]. Koulutuksessa

käytetään puhtaista äänisignaaleista sekä melusta ja kaiusta muodostettuja äänisekoituksia, joista neuroverkko koulutetaan muodostamaan alkuperäiset puhtaat signaalit [15].

2.1.1. Keilanmuodostus

Keilanmuodostuksella (*beamforming*) tarkoitetaan avaruudellista suodatusta, jonka tavoitteena on vahvistaa tietyistä suunnista tulevia signaaleja vaimentamalla ympäröiviä ääniä [13]. Keilanmuodostuksessa eri mikrofonien syötteet suodatetaan ja painotetaan niin, että kiinnostuksen kohteena olevan äänilähteen suuntaan muodostuu herkästi ääntä kuuleva keilamainen alue [16]. Muista suunnista tulevat ympäröivät äänet vaimenevat, jolloin saavutetaan puhtaampi signaali halutusta äänilähteestä. Koska keilanmuodostuksen vaimennus perustuu äänilähteiden eriäviin sijainteihin, se ei toimi hyvin tilanteissa, joissa äänilähteet ovat liian lähellä toisiaan [19].

Keilanmuodostuksen toteutuksen edellytyksenä on useista mikrofoneista koostuva mikrofoniryhmä, jonka muoto tunnetaan [15]. Useiden mikrofonien avulla saadaan informaatiota äänien tulosuunnista (ks. alaluku 2.2), joita käytetään keilan ohjaamiseksi haluttujen äänilähteiden suuntaan [15, 19]. Keilanmuodostusmenetelmät voidaan jakaa kahteen ryhmään: kiinteisiin ja adaptiivisiin [13]. Kiinteässä keilanmuodostuksessa keilan ohjaamiseen käytetyt painokertoimet ja aikaviiveet pysyvät muuttumattomina, kun keila on ohjattu haluttuun suuntaan; adaptiiviset keilanmuodostusmenetelmät taas kykenevät automaattisesti säätämään painokertoimiaan tilanteesta riippuen [13].

Yleisin kiinteä keilanmuodostusmenetelmä on äänisignaalien viiveeseen ja niiden summaamiseen perustuva viivesummain keilanmuodostin (*delay-and-sum beamformer, DSB*) [13]. DSB:ssä kunkin mikrofonin signaaliin lisätään viivettä eriävien saapumisaikojen kompensoimiseksi ja signaalit painotetaan ja summataan konveksina kombinaationa [16]. DSB:tä käytetään yksinkertaisuutensa vuoksi usein lähtökohtana kehittyneemmille keilanmuodostusmenetelmille [16].

Yksi yleisimmistä adaptiivisista keilanmuodostustekniikoista on varianssin minimoiva särötön vaste (*Minimum Variance Distortionless Response, MVDR*) [13]. MVDR-keilanmuodostuksen ideana on pyrkiä muodostamaan suodatin, joka minimoi vasteen varianssin aiheuttamatta säröä tai häiriöitä keilan suunnasta saapuviin signaaleihin [21].

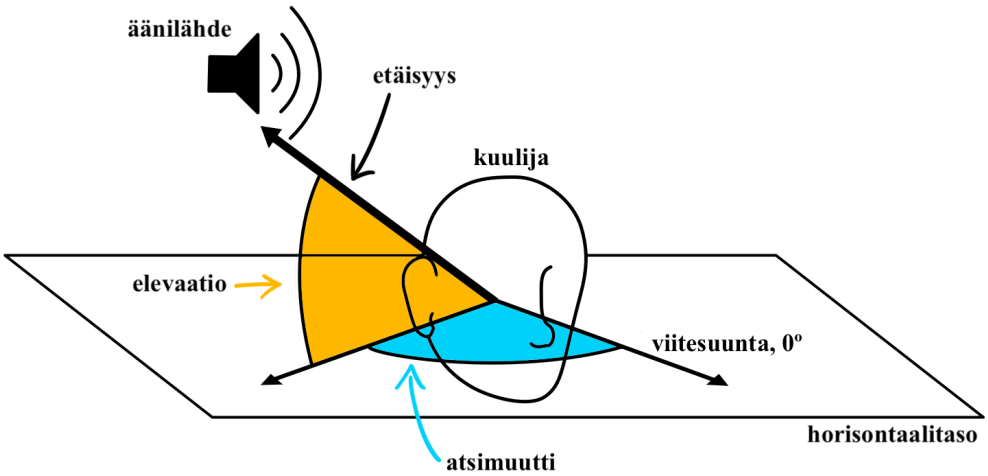
2.2. Äänilähteen sijainnin paikantaminen

Äänilähteidien sijainnin paikallistamisella pyritään selvittämään äänilähteiden sijainteja ympäristössä käyttäen hyväksi yhden tai useamman mikrofonin tallentamia äänisignaalien sekoituksia. Erityisesti konekuulon kiinnostuksen kohteena on äänen tulosuunta kuulijan suhteen (*Direction Of Arrival, DOA*), jonka estimaatteja voidaan käyttää hyväksi esimerkiksi puheentunnistuksessa [22], melun- ja kaiunpoistossa [18] ja äänilähteiden erottelussa (ks. alaluku 2.3). Informaatiota lähteiden sijainneista voidaan hyödyntää myös esimerkiksi kameroiden suuntaamisessa [7] ja kuulolaitteissa [4, 5].

2.2.1. Sijainnin ilmaisu

Äänilähteen sijainti kuulijan suhteen voidaan ilmaista käyttäen atsimuuttia, elevaatiota ja etäisyyttä kuvan 2 mukaisesti. Atsimuutti ja elevaatio ovat astelukuja, jotka kertovat äänilähteen suunnan. Atsimuutti on suunnan kulma viitesuuntaan nähden horisontaalitasossa; elevaatio on puolestaan kulma suuntavektorin ja sen horisontaalitason projektion välillä. Etäisyys kertoo äänilähteen etäisyyden kuulijasta atsimuutin ja elevaation antamasta suunnasta. Nämä kolme suuretta siis muodostavat yhdessä kuulijasta lähtevän vektorin, jonka kärki sijatsee äänilähteessä. [14]

Kaikki suunnan komponentit eivät välttämättä ole tarpeellisia kaikissa käyttökohteissa. Esimerkiksi tieto äänilähteen etäisyydestä voi olla tarpeeton, jolloin järjestelmän tarvitsee estimoida vain suuntia. Suuntaa estimoidessa on myös yleistä arvioida vain suunnan atsimuuttia, jolloin elevaatio voidaan jättää huomiotta.



Kuva 2. Äänilähteen sijainnin ilmaisu käyttäen atsimuuttia, elevaatiota ja etäisyyttä.

2.2.2. Suunnan estimointimenetelmät

Ääntä tallennettaessa usealla mikrofonilla äänisignaalit ja kaiku saapuvat eri mikrofoneihin eri tavoin johtuen mm. mikrofonien sijainneista ja akustisen ympäristön piirteistä, jolloin mikrofonien muodostamiin äänisekoituksiin muodostuu eroavaisuuksia. Eroja syntyy esimerkiksi signaalien viiveessä, amplitudissa ja vaiheessa eri sekoitusten välillä [14, 23]. Tulosuunnan estimointimenetelmät perustuvat usein näiden eroavaisuuksien vertailuun [24]. Koska sijaintidatan käyttökohteilla on usein tiukat vaatimukset reaaliaikaisuudesta, sijaintiestimaattien täytyy olla tarkkoja, nopeasti laskettavissa ja kyetä toimimaan korkealla päivitystaajuudella [25].

Tapoja suuntien estimointiin on useita. Suosittuja menetelmiä ovat mm. signaalien aliavaruuksia hyödyntävä MUSIC-algoritmi (*MUltiple SIgnal Classification*) [26] ja sen johdannaiset [27, 28, 29]. MUSIC-menetelmien etuna on niiden kyky estimoida useampien äänilähteiden suuntia yhtäaikaisesti; samanaikaisten lähteiden määrä voi

olla yhden vähemmän kuin käytettävän mikrofonijärjestelmän mikrofonien määrä [30]. Menetelmien suorituskyky kuitenkin heikkenee huomattavasti, jos ympäristössä on liikaa kaikua [30].

Myös keilanmuodostusta voidaan käyttää hyväksi suuntien estimoinnissa. Esimerkiksi ohjattu vasteteho vaihemuunnoksella (*Steered Response Power with PHAse Transform, SRP-PHAT*) pyrkii määrittämään suuntaestimaatin skannaamalla ympäristöä keilanmuodostuksella toteutetulla keilalla ja maksimoimalla havaitun vasteen tehon [31]. SRP-PHAT:n vahvuutena on sen tehokkuus melusta ja kaiusta huolimatta; sen laskennalliset vaatimukset ovat kuitenkin monissa tilanteissa liian korkeat mikäli skannattava alue on laaja [31, 32].

Yleinen tapa suuntien estimoimiseksi on käyttää äänisignaalien eriäviä saapumisaikoja eri mikrofoneihin eli tuloaikaeroa (*Time Difference Of Arrival*, TDOA) [13, 33]. Tuloaikaeroihin perustuvia estimointimenetelmiä ovat esimerkiksi adaptiivinen ominaisarvohajotelma (*Adaptive Eigenvalue Decomposition*, AED) ja normalisoitu ristikorrelaatio vaihemuunnoksella (*Generalized Cross-Correlation with PHAse Transform*, GCC-PHAT) [22, 34]. TDOA-estimaatit toimivat hyvin hiljaisissa ympäristöissä; haasteena on kuitenkin kaiun ja melun vaikutusten minimointi, joka on ongelmana käytännössä kaikissa käytännön sovelluskohteissa [24].

Äänen tallentamiseen käytettävän mikrofoniryhmän muoto ja dimensiot vaikuttavat siihen, millaista sijaintidataa TDOA-arvoilla on mahdollista kerätä. Esimerkiksi kaksiulotteisia mikrofoniryhmiä käytettäessä, äänien suuntia voidaan arvioida vain mikrofoniryhmän tasossa. Yleensä tämä tarkoittaa atsimuutin estimointia vaakatasossa, sillä pelkkä elevaatio on harvemmin kiinnostava, ellei sovelluskohde sitä vaadi. Kolmiulotteisella pallon muotoisella mikrofoniryhmällä tulosuunnasta voidaan estimoida tarkasti sekä atsimuuttia että elevaatiota [35].

Suuntien estimointiin on viime vuosina kehitetty myös koneoppimista hyödyntäviä ratkaisuja. Esimerkiksi Xiao et al. [36] ja Vesperini et al. [37] ovat pyrkineet paikkaamaan GCC-PHAT-estimaattien heikkoa suorituskykyä meluisissa ja kaikuvissa ympäristöissä syväoppimista käyttäen. Oppimisessa käytettävien neuroverkkojen tarkoituksena on oppia piirteitä häiriöitä sisältävistä ristikorrelaatiokuvioista, jotka tuottavat tietynlaisia DOA-arvoja [36].

Myös SRP-PHAT-menetelmän korkeita vaatimuksia on pyritty keventämään syväoppimismenetelmillä: esimerkiksi Diaz-Guerra et al. [32] ovat muuntaneet suuntaestimoinnin regressio-ongelmaksi, jonka kompleksisuus kasvaa tavanomaista SRP-PHAT-menetelmää hitaammin skannausalueen kasvaessa.

Wang et al. [38] ovat puolestaan käyttäneet syväoppimisella estimoituja aika-taajuus-maskeja parantaakseen ristikorrelaatioon, keilanmuodostukseen ja aliavaruuksiin perustuvien menetelmien melun- ja kaiunsietoa. Heidän kokeelliset tuloksensa osoittavat syväoppimisen parantavan tavanomaisten menetelmien suorituskykyä huomattavasti.

Vaikka suuntien estimointiin yleensä käytetään useiden mikrofonien ryhmiä, estimointi on mahdollista myös monauraalisesti, eli vain yhtä mikrofonia käyttäen [11, 12]. Monauraaliset menetelmät ovat kuitenkin huomattavasti useiden mikrofonien menetelmiä epätarkempia [23], eivätkä näin ollen ole kovin suosittuja.

2.2.3. Binauraalinen suunnan määritys

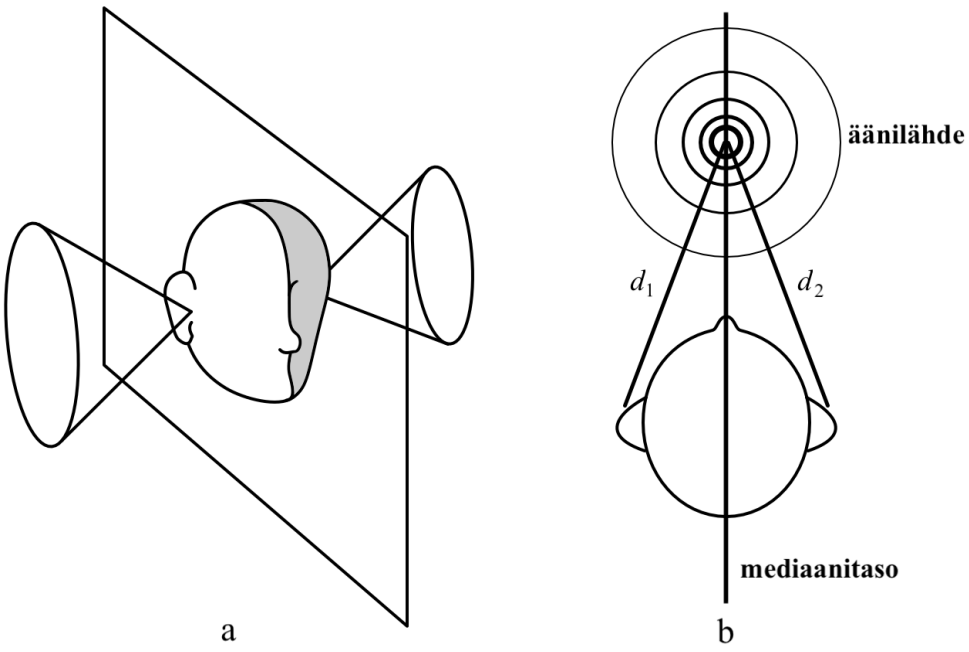
Binauraalisella kuulolla tarkoitetaan kuulemista käyttäen kahta korvaa, tai mikrofonia. Koska ihmisen kuulo on binauraalinen, jotkin binauraaliset suunnanmääritysmenetelmät pyrkivät toteuttamaan suunnan estimoinnin ihmisen kuulon tavoin [39]. Yksi binauraalisten menetelmien käyttökohde onkin kuulolaitteet [4, 5], joissa laitteiden suorituskykyä melussa pyritään parantamaan keilanmuodostuksen avulla.

Kun äänen sijaintia määritetään binauraalisesti, tärkeitä mitattavia tekijöitä ovat ääniaaltojen intensiteetin (*Interaural Level Difference, ILD*) ja ajan (*Interaural Time Difference, ITD*) erot korvien välillä [14, 40]. ILD ja ITD perustuvat siihen, että äänilähdettä lähinnä oleva korva (*ipsilateraalinen*) kokee suuremman intensiteetin kuin kauempi korva (*kontralateraalinen*) ja ääni saapuu lähempään korvaan kauempaa korvaa aiemmin.

Näiden suureiden käytössä ilmenee kuitenkin ongelmia silloin, jos äänilähde sijaitsee kuvan 3 mukaisesti joko mediaanitasolla, jolloin molemmat korvat ovat yhtä kaukana äänilähteestä, tai pään jommalla kummalla sivulla niin sanotussa sekaannuskartiossa [14, 39, 41]. Sekaannuskartioiden ongelmana on se, että samat ITD-arvot voivat vastata useita eri suuntia, jolloin tarkkoja suuntia ei voida määrittää [41]. Tästä johuten monet konekuulojärjestelmät pyrkivät käyttämään toteutuksissaan useampaa kuin kahta mikrofonia [42].

Yksi tapa määrittää äänilähteen suunta mediaanitasolla tai sekaannuskartioissa on pään kääntäminen [5, 41]. Kääntäminen aiheuttaa eroavaisuuksia korvien etäisyyksissä ja asennossa suhteessa äänilähteisiin, jolloin lähteiden sijainnit voidaan määrittää käyttäen ILD ja ITD -arvoja.

Toinen tapa on hyödyntää äänisignaalien spektriä, eli pään siirtofunktiota (*Head Related Transfer Function, HRTF*). Siirtofunktio saa arvoja sen perusteella, miten ääni saapuu korvaan; arvot ovat siis vahvasti riippuvaisia pään ja korvien muodosta ja äänilähteen sijannista [14, 42]. Eri korvien HRTF-arvot eroavat myös mediaanitasolla ja sekaannuskartioissa, jolloin niitä voidaan käyttää suunnan määritykseen. HRTF on myös keskeisin tapa estimoida äänilähteiden elevaatioita [43]. Siirtofunktio muodostetaan käyttäen tietoa pään ja korvien muodoista, jonka vuoksi se täytyy kyetä laskemaan uudelleen mikäli pään ja korvien muodossa tapahtuu muutoksia, esimerkiksi kun päähän asetetaan hattu [42].



Kuva 3. (a) Pään halkaiseva mediaanitaso ja sivuilla sijaitsevat sekaannuskartiot. (b) Äänilähteen sijaitessa mediaanitasolla, korvien ja äänilähteen väliset etäisyydet d_1 ja d_2 ovat yhtä suuret.

2.3. Äänilähteiden erottelu

Tehokkaan konekuulojärjestelmän toteuttaminen edellyttää ratkaisua cocktailkutsuongelmaan (*cocktail party problem*) [10]. Todenmukaisessa akustisessa ympäristössä, kuten esimerkiksi cocktailkutsuilla, on tyypillistä, että samanaikaisesti puhuvia ihmisiä on useita ja taustahälyä on paljon, esimerkiksi musiikin muodossa. Cocktailkutsuilmiössä on kyse ihmisen kyvystä kuulla tällaisessa ympäristössä, eli erottaa ja keskittyä yhteen äänilähteeseen muiden kilpailevien äänilähteiden ja taustamelun seasta [43, 13].

Puheenerottelu on vain ensimmäinen askel coctailkutsuilmiön ratkaisemisessa. Kun päämääränä on ihminen-kone vuorovaikutuksen edistäminen, on myös automaattisella puheentunnistuksella tärkeä rooli. Toisin kuin ihmiset, jotka voivat keskittyä pääasiassa vain yhteen puhujaan kerrallaan, voivat koneet käsitellä useita puhujia yhdenaikaisesti, jos puhujat on jaettu eri kanaviin. Puhujien erottelemisessa koneet eivät kuitenkaan vielä yllä ihmisen tasolle.

Useiden mikrofonien kokoelmien yleistyessä monikanavaiset avaruudellista suodatusta käyttävien puheenerottelumenetelmien merkitys tulee kasvamaan [13], mutta myös yksikanavaisen puheen erotteleminen on yhä välttämätöntä. Monissa tallennuslaitteissa on yhä vain yksi sisäänrakennettu mikrofoni, eivätkä useampaa mikrofonia käyttävät avaruudellisen suodatuksen tekniikat kykene erottelemaan puhujia kahden henkilön puhuessa samasta suunnasta [13]. Viime vuosina myös

syväoppimisen keinoja on esitetty cocktailkutsuilmiön ratkaisemiseksi ja niiden osalta suurin osa tutkimuksesta liittyy yksikanavaisen puheenerottelun ongelmiin [13].

Nykyiset puheenerottelumenetelmät ovat jakautuneet sen perusteella hyödyntävätkö ne koneoppimista [44]. Koneoppimiseen perustuvien menetelmien etuina ovat parempi luokittelutarkkuus ja parempi suorituskyky häiriöpitoisten signaalien erottelussa muihin menetelmiin verrattuna; hintana on kuitenkin suurempi laskennallinen monimutkaisuus [44]. Tietokoneiden suorituskyvyn kasvaessa tämän rajoituksen merkitys tulee kuitenkin vähenemään.

2.3.1. Yksikanavainen puheenerottelu

Puheen erotteleminen yksikanavaisesta äänitallenteesta on ollut haastava ongelma jo useiden vuosikymmenien ajan [45, 46]. Yksikanavaisissa, eli vain yhdestä mikrofonista tallennettujen signaalien erottelussa äänen tulosuuntia ei voida havainnoida ja voidaan käyttää vain puheen ja taustaäänien luontaisia ominaisuuksia [47].

CASA

Koska puheen erottelu on niin helppoa ihmisille, yksi selkeä lähestymistapa on pyrkiä mallintamaan ihmisten tapaa erotella puhetta. Ihmisen kuulojärjestelmän äänien erottelutavan uskotaan toimivan kuulema-analyysi-mallin (*Auditory Scene Analysis, ASA*) mukaisesti [48]. Koska ASA toimii tehokkaasti, pyritään sitä mallintamaan tietokoneella laskennallisen kuulema-analyysin (*Computational Auditory Scene Analysis, CASA*) avulla [49]. Vaikka CASA on esitetty yli vuosikymmen sitten, samaan periaatteeseen pohjautuvia tekniikoita kehitetään yhä [13].

CASA-pohjaiset ratkaisut ovat osoittautuneet tehokkaiksi puheen erottelussa, mutta niillä on myös merkittäviä rajoitteita [50]. CASA:n puheen erottelu rajoittuu vain soinnilliseen puheeseen [13, 50, 46]. Soinnuttomat puheen osat, kuten äänteet p, t, k, tai s, voivat sekoittua taustaääniin, koska niiltä puuttuu harmoninen rakenne ja niiden energia on heikkoa [51, 50]. CASA:n tarkkuus on myös riippuvainen äänen sävelkorkeuden mittauksen tarkkuudesta [13].

Ei-negatiivisien matriisien tekijöihin jako

Ei-negatiivisien matriisien tekijöihin jako (*Non-negative Matrix Factorization, NMF*) [52] perustuu oletukseen äänen spektrogrammin alhaisen asteen rakenteesta, joka voidaan esittää pienenä määränä kantoja [13]. NMF:ssä:

$$Y = \sum_{s} W_s H_s, \tag{2}$$

jossa jokaisen äälilähteen s alhaisen asteen likiarvo on mallinnettu ei-negatiivisilla matriiseilla W_s ja H_s ja summattu muodostamaan sekoittunut signaali Y [13]. Hajotelmamatriisien ei-negatiivisuudesta johtuen lähteet eivät supista toisiaan sekoitteen Y uudelleenrakentamisessa [13]. NMF:n koulutusvaiheessa jokainen

puhdas äänilähde, kuten puhe, melu ja musiikki hajotetaan, ja sille muodostetaan lähdekohtainen hakemisto W. Testausvaiheessa kaikista lähdekohtaisista hakemistoista muodostetaan yhdistetty kiinteä hakemisto, ja vain aktivaatiokertoimet H optimoidaan jokaiselle lähteelle. Yksinkertainen NMF algoritmi on

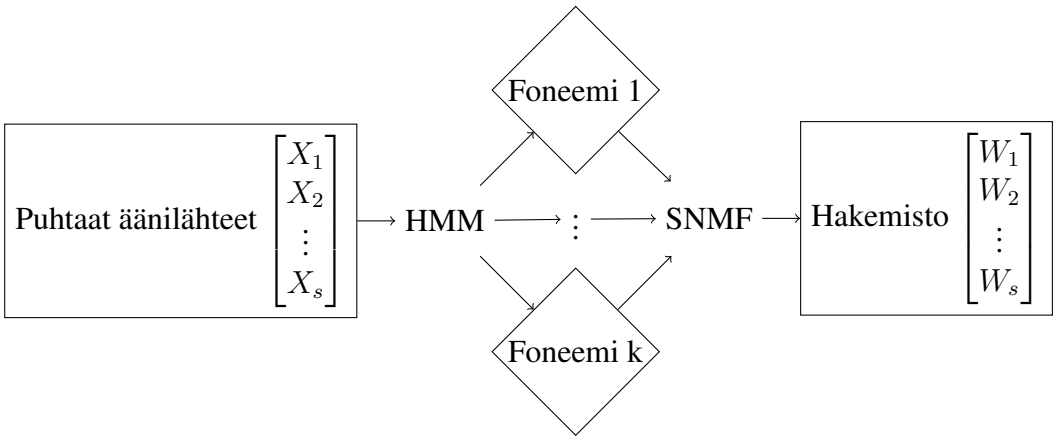
$$min_{W,H} D(Y||WH) : W, H \geq 0, \tag{3}$$

jossa $D(Y||WH)$ on [52] esitetty kustannusfunktio.

NMF:ää voidaan käyttää äänilähteiden erotteluun yksikanavaisesta syötteestä, sekä kohinanpoistoon puhtaan puhesignaalin estimoimiseksi [53]. NMF-menetelmistä on esitetty useita variaatioita, kuten harva NMF [54, 55], joka pakottaa kerroinmatriisin H olemaan harva. Termillä harva viitataan signaalimalliin, jossa data esitetään pienellä määrällä suuremmasta datajoukosta valittuja aktiivisia elementtejä [55]. Vakaa NMF [56] pyrkii minimoimaan ei-negatiivisen matriisin Y ja sen rekonstruktion välisen β-divergenssin [57]. Konvoluutionaalisessa NMF:ssä [58, 56] spektrogrammi hajotetaan kannan ja aktivaation konvoluutioksi (tulon sijaan).

Kuvassa 4 on esitetty kaksi menetelmää hakemistojen oppimiseen äänilähteistä. Yksi menetelmä (a) on oppia hakemisto koko koulutusjoukosta. Toinen menetelmä (b) on segmentoida koulutusjoukon äänilähteet yksittäisiksi foneemeiksi, oppia väljä hakemisto jokaiselle foneemille ja koota hakemisto ketjuttamalla yksittäisille foneemeille opitut hakemistot.

a



b

Kuva 4. NMF oppimismalleja.

NMF:n onnistumista rajoittaa sen kannat W; muita puhesignaalien ominaisuuksia ja säännönmukaisuuksia ei huomioida [13]. Sen lisäksi jokaisella eroteltavalla äänilähteellä täytyy olla opittu hakemisto, eli äänilähteen tulee olla osa koulutusjoukkoa, joka ei ole mahdollista monissa käytännön toteutuksissa [13].

2.3.2. Monikanavainen puheenerottelu

Monikanavaisten puheenerottelumenetelmien etu yksikanavaisiin menetelmiin verrattuna on niiden kyky hyödyntää signaalien avaruudellista informaatiota erottelun toteuttamisessa [59, 60]. Useiden mikrofonien avulla saavutetaan esimerkiksi tehokkaampi melun- ja kaiunpoisto [18], joka helpottaa signaalien erottelua.

Monikanavaiset puheenerottelumenetelmät voidaan jakaa keilanmuodostusta hyödyntäviin menetelmiin ja sokean lähteiden erottelun menetelmiin [13]. Keilanmuodostusmenetelmät suodattavat signaalit ensin monikanavaisella keilanmuodostuksella ja käyttävät valittua yksikanavaista puheenerottelumenetelmää suodatetun signaalin analysointiin. Sokean lähteiden erottelun menetelmät toteuttavat erottelun vertailemalla eri mikrofonien äänisekoituksia sillä oletuksella, että sekoitukset muodostavat äänilähteet ovat toisistaan riippumattomia [61].

Sokea lähteiden erottelu

Sokean lähteiden erottelun (*Blind Source Separation, BSS*) menetelmät ovat olleet 1980-luvulta alkaen aktiivisen tutkimuksen kohteena useiden eri tieteenalojen, kuten signaalinkäsittelyn, tilastotieteen ja neuroverkkojen toimesta [62]. Äänilähteiden erottelun tapauksessa menetelmiä voidaan käyttää erotteluongelman ratkaisuun, kun tallennettu äänisignaali muodostuu useiden erillisten äänilähteiden sekoitteesta [63]. Tyypillisesti analyysin kohteena ovat useista mikrofoneista saatavat äänisekoitteet, jotka koostuvat erilaisista alkuperäisten äänilähteiden muodostamista kombinaatioista [61]. Erottelumenetelmät pyrkivät hyödyntämään sekoitusten välistä korreloivaa dataa erottelun toteuttamiseksi [13].

Termillä "sokea" viitataan siihen, että sekoitusten muodostavat alkuperäiset lähdesignaalit eivät ole tiedossa, eikä sekoituksista ole saatavilla minkäänlaista ennakkoinformaatiota [61, 64]. BSS-menetelmien suurin etu onkin kyky suoriutua vaikka informaatiota on rajallisesti. Tiedon puutetta kompensoidaan oletuksella, että lähdesignaalit ovat toisistaan riippumattomia [61]. Sekoittuneita signaaleja käsitellään riippumattomien lähdesignaalien lineaarikombinaatioina [65].

Yksi yleinen sokean lähteiden erottelun menetelmä on riippumaton komponenttianalyysi (*Independent Component Analysis, ICA*). Riippumattoman komponenttianalyysin menetelmät perustuvat tuntemattomien äänilähteiden tilastolliseen riippumattomuuteen [62]. Jokaisen riippumattoman komponentin jakauman myös oletetaan olevan epägaussinen; jakaumien tarkkaa muotoa ei kuitenkaan tarvitse tuntea [66]. Riippumattoman komponenttianalyysin mallin (*ICA model*) mukaan äänisekoituksissa $\mathbf{x}$ alkuperäiset lähteet ovat sekoittuneet lineaarisesti

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{4}$$

jossa $\mathbf{A}$ on kääntyvä neliömatriisi, joka sekoittaa tuntemattomat lähteet $\mathbf{s}$ [66, 67]. Tarkoituksena on selvittää sekoitusmatriisin rakenne käyttäen edellä esiteltyjä oletuksia, jolloin on mahdollista laskea sekoitusmatriisin käänteismatriisi [66]. Tämän jälkeen yhtälöstä (4) voidaan ratkaista alkuperäisten äänilähteiden estimaatit

$$\hat{\mathbf{s}} = \mathbf{W}\mathbf{x} \tag{5}$$

jossa $\mathbf{W}$ on sekoitusmatriisin käänteismatriisin $\mathbf{A}^{-1}$ approksimaatio [66, 67].

2.3.3. Syväoppimismenetelmät

Viime vuosina syväoppimismenetelmiä on esitetty myös cocktailkutsuilmiön ratkaisemiseen niiden hyvän menestyksen myötä puheentunnistuksen saralla [13]. Syväoppimismallit ovat tehokkaita pääasiassa silloin kun puheen erottelu voidaan laatia ohjatun oppimisen ongelmaksi[13]. Koska saman sekoittuneen signaalin voi muodostaa ääretön määrä mahdollisia lähdesignaali yhdistelmiä, on tarpeellista oppia puhesignaalien säännönmukaisuuksia koulutusjoukosta, jotta mahdottomat yhdistelmät voidaan poissulkea [13].

Yleiset ongelmat

Syväoppimismenetelmiä hyödynnettäessä puheenerottelussa kohdataan usein kaksi ongelmaa: permutaatio-ongelma ja tuloksien dimensionaalisuuden epäsuhta [68, 69].

Suurin osa neuroverkkomenetelmistä koulutetaan kartoittamaan syötesignaali uniikkiin kohdetulosteeseen, joka voi olla nimiö, jono tai regressioviite [68]. Koulutuksen aikana täytyy laskea erotellun signaalin ja sitä vastaavan puhtaan viitesignaalin välinen virhe [70]. Permutaatio-ongelma puheentunnistuksessa syntyy siitä, että tunnistuksen kohteiden järjestystä puhesekoitteessa ei voida määrittää [13, 68, 71, 72]. Tämä tarkoittaa sitä, ettei koulutusvaiheessa voida ennalta tietää, mihin viitesignaaliin eroteltua signaalia tulisi verrata. Ongelma voidaan ratkaista naiivisti valitsemalla tietty järjestys, jos puhujia on vähän ja äänet ovat selkeästi erotettavissa toisistaan, esimerkiksi mies- ja naispuhuja [71]. Ongelmaan on myös esitetty viime aikoina useita ratkaisuja, kuten syväklusterointi (*Deep Clustering, DC*) [72] ja permutaatioinvariantti koulutus (*Permutation Invariant Training, PIT*) [71].

Tuloksien dimensionaalisuuden epäsuhdalla viitataan ongelmaan, joka kohdataan käytettäessä neuroverkkoja, joiden tulossolmujen lukumäärä on vakio. Tällaiset verkot eivät pysty käsittelemään luotettavasti tilanteita, joissa eroteltavien signaalien lukumäärä vaihtelee [68, 69].

Syväklusterointi

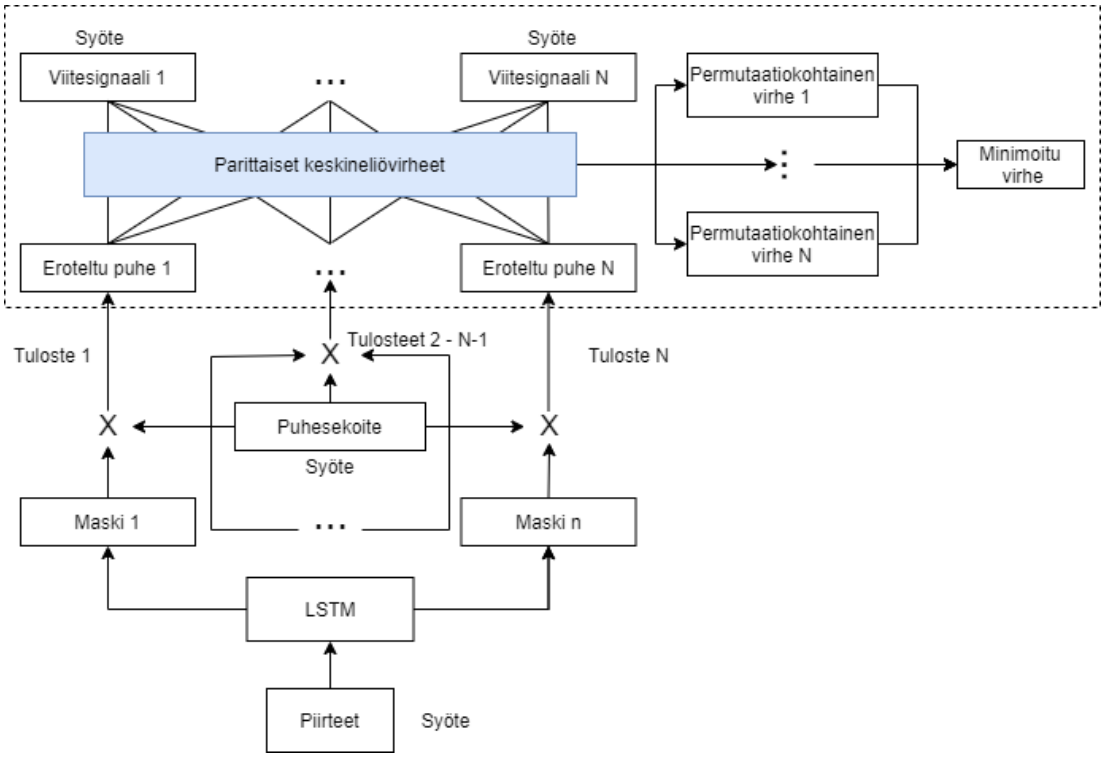
Hershey et al. [72] ehdottavat syväklusteroinniksi kutsuttua kehystä permutaatio-ongelman ratkaisemiseen. Syväklusterointi pyrkii ratkaisemaan cocktailkutsuilmiön kouluttamalla neuroverkon esittämään jokainen aika-taajuus yksikkö (t, f) korkean dimension upotteiksi niin, että saman puhujan hallitsemien aika-taajuus yksikköjen upotteet ovat lähellä toisiaan, ja eri puhujien hallitsemat aika-taajuus yksikköjen

upotteet ovat kauempana toisiaan [60]. Näin puhuja voidaan määritellä jokaiselle aika-taajuus yksikölle soveltamalla yksinkertaista klusterointimenetelmää opittuihin upotteisiin, kuten k-means [73].

Permutaatioinvariantti koulutus

Kolbæk et al. [71, 70] esittävät permutaatio-ongelman ratkaisuksi permutaatioinvarianttia koulutusta. Toisin kuin tavanomaisessa puheenerottelumallissa, viitesignaalit käsitellään kokoelmana järjestetyn listan sijaan. Toisin sanoen menetelmällä saavutetaan sama koulutustulos viitesignaalien järjestyksestä riippumatta. Tämä ominaisuus saavutetaan PIT:llä määrittämällä ensin kaikki mahdolliset yhdistelmät viitesignaalien ja yhdistelmäsignaalista estimoitujen lähdesignaalien välillä. Mahdollisten yhdistelmien lukumäärä on lähteiden lukumäärän kertoma $N!$. Tämän jälkeen kaikille yhdistelmille lasketaan permutaatiokohtainen virhe, joka koostuu yhdistelmän viitesignaalien ja estimoitujen lähdesignaalien parittaisten keskineliövirheiden *(Mean Squared Error, MSE)* summasta. Pienimmän keskineliövirheen permutaatio valitaan, ja malli optimoidaan minimoimaan tämä pienin keskineliövihe. Malli siis toteuttaa yhdenaikaisesti estimoitujen lähteiden nimiöimisen ja virheen arvioinnin.[71, 70]

Kuvassa 5 on esitetty PIT:n koulutuksen kulku N:n puhujan erottelumallissa jokaiselle otokselle, joka sisältää viitesignaalit 1-N, niistä muodostetun puhesekoitteen, sekä puhesekoitteesta lasketut piirteet.



Kuva 5. N:n puhujan puheenerottelumalli permutaatioinvariantilla koulutuksella.

PIT skaalautuu hyvin useiden puhujien erotteluun. N:n puhujan mallissa tarvitsee laskea vain N^2 parittaista neliösummaa, jotta kaikki $N!$ permutaatiokohtaista virhettä

voidaan laskea. Koska $N!$ kasvaa paljon nopeammin puhujien määrästä riippuen kuin N^2, ja parittaisten keskineliösummien määrittämisen vaatima laskentateho on paljon suurempi kuin permutaatiokohtaisen virheen, joka on vain kyseisen permutaation parittaisten keskineliövirheiden summa, mallin suorituskyky kestää puhujien määrän kasvamista hyvin.[70]

Puheenerottelua suoritettaessa saatavilla on pelkästään sekoittunut puhe. Erottelu suoritetaan jokaiselle syötenäytteelle, joista estimoidaan tuloste. PIT:n koulutuksen vertailuperusteiden vuoksi permutaatio pysyy samana kaikilla näytteillä saman tulosteen sisällä, mutta permutaatiot voivat muuttua tulosteiden välillä [70]. Yksinkertaisimmassa asetelmassa voidaan tehdä oletus, että permutaatiot eivät muutu tulosteiden välillä, kun puhujia erotellaan [70]. Kolbæk et al. [71] kuitenkin osoittavat tämän johtavan heikkoihin tuloksiin. Parempien tulosten saavuttamiseksi PIT:n lisäksi tarvitaan puhujanerottelualgoritmeja, jotka määrittävät tulosteiden permutaatiot puhujat huomioon ottaen [70].

Lausahdustason permutaatioinvartiantti koulutus

Yksi ratkaisu tulosteiden välisten permutaatioiden muunnosten havainnointiin, eli niin sanottuun jäljennysongelmaan, on vertailla keskineliövirheitä maskien eri permutaatioilla peräkkäisten tulosteiden päällekäisissä osuuksissa. Ratkaisu vaatisi kuitenkin erillisen jäljitysvaiheen, joka voi monimutkaistaa mallia. Toisekseen, kun jälkimmäisten otosten permutaatio riippuu aiempien otosten permutaatiosta, yksikin virhe aiemmissa otoksissa muuntaisi sen jälkeisille otoksille määritettävää permutaatiota. [70]

Kolbæk et al. [70] esittävät ongelman ratkaisuksi lausahdustason permutaatioinvartianttia koulutusta *(utterance-level Permutation Invariant Training, uPIT)*, joka ratkaisee sekä jäljennysongelman, että permutaatio-ongelman paremmin kuin alkuperäinen PIT [70]. Alkuperäisessä PIT:ssä optimaalinen permutaatio lasketaan jokaiselle tulosteelle erikseen, kun taas uPIT käyttää samaa, lausahdustasolla erotteluvirheen minimoivaa permutaatiota kaikille tulosteille samassa lausahduksessa [70]. Toisin sanottuna kuvassa 5 esitetyt keskineliövirheet lasketaan koko lausahdukselle olettaen, että kaikki otokset lausahduksen sisällä noudattavat samaa permutaatiota [70].

Koska lausahduksien pituudet vaihtelevat ja tehokas erottelu oletettavasti vaatii signaalien riippuvuuksien hyödyntämistä pitkältä aikaväliltä, malli käyttää pitkän lyhytkestomuistin *(Long Short-Term Memory, LSTM)* takaisinkytkeytyvää neuroverkkoa *(Recurrent Neural Network, RNN)*. Syvän LSTM:n avulla lausahduksen otokset evaluoidaan koko historiatietoa hyväksikäyttäen jokaisella tasolla. Puheen erottelua suoritettaessa uPIT:llä ei tarvitse laskea parittaisia keskineliövirheitä kaikille mahdollisille permutaatiolle, vaan oletetaan permutaation pysyvän samana kaikilla otoksilla samassa lausahduksessa. Tämä tekee uPIT:stä yksinkertaisen ja kiinostavan ratkaisun. [70]

2.4. Hahmontunnistus

Automaattinen puheentunnistus (*Automatic Speech Recognition, ASR*) on yksi keskeisimmistä menetelmistä joilla koneet voivat simuloida ihmistä. Ihmisten välinen puhekommunikaatio on kuitenkin erittäin monimutkaista ja siihen liittyy useita opinaloja, kuten akustiikka, fonetiikka, kielitiede ja psykologia [74]. Jokaisella ihmisellä on erinlainen ääntöväylä ja luontainen tapa puhua. Puhesignaaleissa on merkittävästi puhujien välistä vaihtelua johtuen esimerkiksi puutteellisesta kielitaidosta (muuna kuin äidinkielenään kieltä puhujat, lapset, yms.), joka voi johtaa merkittäviin eroavaisuuksiin kielen sääntöjen mukaisesta rakenteesta, sekä puhujan sisäistä vaihetelua johtuen esimerkiksi puhujan tunne-, tai terveydentilan muutoksista [75].

Automaattinen puheentunnistus pyrkii tunnistamaan kuvioita syötetyistä ääniaalloista, yleisin tavoite on kääntää puhesignaali tekstiksi, eli muodostaa tekstitietoa siitä, mitä puhuja on sanonut [74]. Automaattisen puheentunnistuksen suurin haaste [74] on puhesignaalien laaja vaihtelu [75], joka tuottaa suuren määrän "hyväksyttäviä" lausahduksia, joita suurin osa ihmiskuulijoista osaisi tulkita. Esimerkiksi jos automaattinen puheentunnistus on kehitetty vain yhden puhujan puhesignaaleja käyttäen, sen tulokset voivat olla tarkkoja samalla puhujalla, mutta jos puhuja tai käytetty mikrofoni muuttuu, sen tarkkuus todennäköisesti heikkenee. Automaattisen puheentunnistuksen tarkkuus on siis riippuvainen opetusaineiston ja testausaineiston välisestä empiirisestä samankaltaisuudesta [74].

Suuren sanavaraston puheentunnistuksessa puheen rakenneosien, kuten tavujen tai foneemeiden käyttö on lähes välttämätöntä, sillä olisi hyvin vaikeaa kerätä riittävä joukko opetusaineistoa sanoja, tai jopa suurempia yksikköjä käyttävien kätkettyjen Markovin mallien sunnittelemiseen. Erikoistuneissa käyttötarkoituksissa kun sanavarasto on pieni, on kuitenkin sekä järkevää, että käytännöllistä käyttää sanoja puheentunnistuksen rakenneosina. [76]

Kaupallisten pilvilaskentaa hyödyntävien ratkaisujen, kuten Google Assistantin, Applen Sirin tai Amazonin Alexan lisäksi puheentunnistusta varten on saatavilla myös useita avoimen lähdekoodin ratkaisuja [77]. Paikallisesti suoritettava puheentunnistus häviää monissa tapauksissa pilvilaskennalle tehokkuudessa, mutta se ei vaadi Internet-yhteyttä, eikä puhetta tarvitse näin ollen myöskään välittää laitteen ulkopuolelle.

3. TOTEUTUS

Työ toteutettiin osana sulautettujen ohjelmistojen projektikurssia, jonka tavoitteena oli toteuttaa toimiva InMoov-robotti, joka sisältää konenäön, konekuulon, puhesynteesin, ja pään liikuttelun komponentteja. Projektissa toteutettiin tarvittava konekuulon osakomponentti, joka yhdistää monikanavaisen mikrofonijärjestelmän ja yksikanavaisen puheenerottelun erotellakseen puhujat mahdollisimman selkeästi automaattista puheentunnistusta varten meluisassa ympäristössä, ja useiden puhujien ollessa yhdenaikaisesti äänessä. Järjestelmä välittää myös arvion puheen tulosuunnasta robotin pään kääntämistä varten.

Toteutuksessa pyrittiin selvittämään, kuinka hyvin uPIT:llä koulutettu syväoppimismalli täydentää konekuulojärjestelmää yksikanavaisella puheenerottelulla. Lisäksi, tavoitteena on löytää menetelmä äänilähteiden suuntien havainnointiin robotin pään kääntämiseksi.

3.1. Kehitysalusta

Työssä käytettävä InMoov-robotti on ranskalaisen suunnittelijan ja kuvanveistäjän Gaël Langevinin kehittämä avoimen lähteen 3D-tulostettava humanoidirobotti. Robottia on kehitetty vuodesta 2012 Langevinin ja InMoovin ympärille muodostuneen yhteisön toimesta ja se on monien yliopistojen, laboratorioiden ja harrastelijoiden käytössä. [78]

ROS (*Robot Operating System*) on avoimen lähdekoodin alusta robottien ohjelmointia varten. Se tarjoaa puitteet robottien ohjelmointiin ja helpottaa robottien koodien siirtämistä robotista toiseen. Useat yritykset ja tutkimuslaitokset käyttävät nykyään ROS:ia toteutuksissaan. [79]

Toteutus on riippuvainen seuraavista Python-moduuleista:

- PyTorch [80],
- LibROSA [81],
- SciPy [82],
- Rospy [83],
- PyAudio [84], ja
- PyYAML.

Merkittävimpiä kirjastoriippuvuuksia ovat PyTorch, LibROSA, SciPy ja Rospy. PyTorch on avoimen lähdekoodin koneoppimisen kehysympäristö, jota käytetään neuroverkon kouluttamiseen ja yksikanavaiseen puheenerotteluun. LibROSA tarjoaa audioanalyysin menetelmiä, ja sitä käytetään lyhytaikaisten Fourier-muunnosten laskemiseen. SciPy toimii laskennallisena kehysympäristönä, ja sitä hyödynnetään myös WAV *(Waveform Audio File Format)* audiotiedostojen käsittelyssä. Rospy tarjoaa ROS-ohjelmointirajapinnan Pythonille.

3.2. Mikrofonijärjestelmä

Yksi työn suunnitteluvaiheessa esiin nousseista kysymyksistä oli sovelluskohteeseen sopivan mikrofoniratkaisun löytäminen. Alustavassa suunnitelmassa äänen tallentamiseen oli tarkoitus käyttää kahta InMoov-robotin silmissä sijaitsevaa mikrofonia. Taustatutkimuksen perusteella silmien mikrofonien hyödyntämisestä löytyi kuitenkin puutteita: Vain kahden mikrofonin käyttäminen tässä käyttötarkoituksessa vaikutti epävarmalta ratkaisulta, sillä luotettavien estimaattien laskeminen äänien tulosuunnille on vaikeampaa kahdella mikrofonilla useiden mikrofonien ryhmiin verrattuna (ks. alaluku 2.2.3). Etenkin mikrofonien sijainti robotin kasvoissa herätti kysymyksiä, sillä paikantamisessa tulisi ottaa huomioon robotin pään muodostama akustinen varjo, joka hankaloittaisi suuntien laskemista.

Kahden mikrofonin ratkaisu päätettiin korvata jonkinlaisella usean mikrofonin ryhmällä. Koska toteutuksessa ei ole tehty ennakko-oletuksia äänien atsimuuttisuunnista, mikrofoniratkaisun tulisi tallentaa ääntä mahdollisimman samalla tavalla horisontaalitasossa suunnan atsimuutista riippumatta. Korkeudesta estimaatteja ei tässä käyttötapauksessa tarvita, joten kolmiulotteiset mikrofonijärjestelmät voitiin siis sulkea pois.

Tilanteissa, joissa mielenkiintoiset äänet voivat kuulua mistä atsimuuttisuunnasta tahansa, ympyrän muotoiset mikrofonijärjestykset ovat etuasemassa muihin järjestyksiin verrattuna niiden symmetrisyyden vuoksi. Symmetrisyys yksinkertaistaa äänisignaalien käsittelyä, kun tietyistä suunnista tulevia ääniä ei tarvitse käsitellä eri tavoin kuin toisista [85]. Ideli et al. [18] ovat myös osoittaneet ympyrän muotoisen usean mikrofonin ryhmän olevan suorituskyvyltään huomattavasti parempi suoraan mikrofonilinjaan verrattuna.

Työn toteutukseen otettiin käyttöön kuvassa 6 näkyvä ympyrän muotoinen neljästä mikrofonista koostuva Seeed Studion ReSpeaker Mic Array v2.0 -mikrofonijärjestelmä[1]. Mikrofoni valittiin sen kattavien ominaisuuksien ja edullisen hinnan vuoksi; myös valmistajan tuotesivulla antamat esimerkkiäänitteet vaikuttivat lupaavilta. Mikrofonin kerrottiin myös soveltuvan käytettäväksi ääniohjattavissa roboteissa ja että sitä on käytetty ROS-järjestelmissä [86].

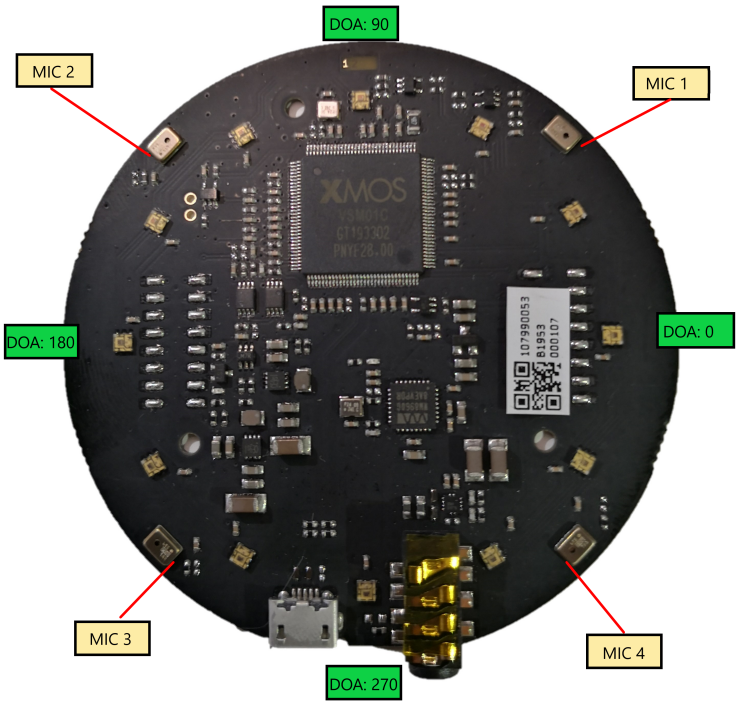
Järjestelmän yhteydessä toimiva XMOS XVF-3000 -piirisarja sisältää käyttövalmiina useita puheenerottelun ja -tunnistuksen kannalta hyödyllisiä ominaisuuksia, kuten

- akustisen suunnanmäärityksen,
- adaptiivisen keilanmuodostuksen,
- melun- ja kaiun vaimennuksen ja
- automaattisen vahvistuksen säädön.

Näitä ominaisuuksia hyödynnetään puheenerottelualgoritmille syötettävien äänisignaalien puhdistamiseksi. Suunnanmäärityksen antama DOA-arvo julkaistaan luotuun ROS-aiheeseen muiden komponenttien saataville. Mikrofonilla voidaan

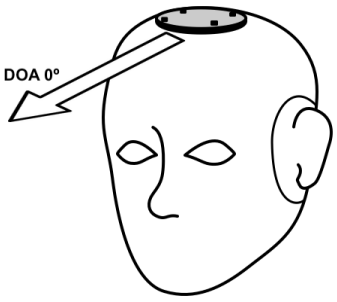
[1] Valmistajan tuotesivu: `www.seeedstudio.com/ReSpeaker-Mic-Array-v2-0.html`

tallentaa ääntä noin viiden metrin säteellä, jonka tulisi olla tarpeeksi tässä käyttötarkoituksessa.



Kuva 6. ReSpeaker Mic Array v2.0[2]

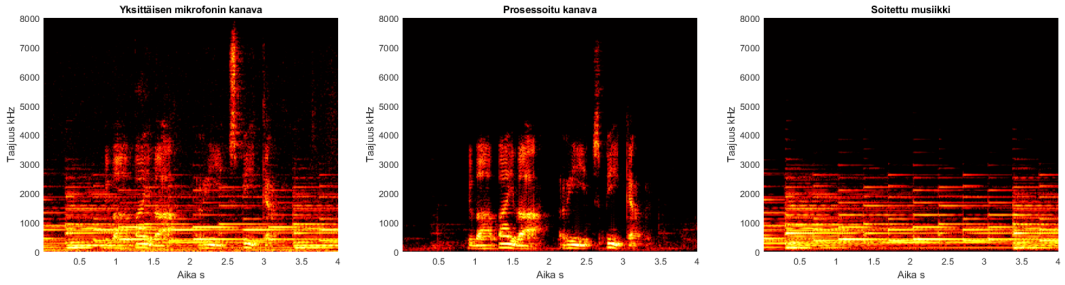
Mikrofonijärjestelmä kiinnitetään vaakatasossa robotin päälaelle kuvan 7 mukaisesti, jotta eri suunnista saapuvat äänet voidaan havannoida mahdollisimman tarkasti. Päähän kiinnitetty mikrofoni myös kääntyy robotin pään mukana, jolloin mikrofoniin asetettu suunnan nollataso säilyy aina robotin katsesuunnassa.



Kuva 7. Mikrofonin sijainti InMoov-robotin päälaella.

Kuva 8 havainnollistaa kuinka mikrofonijärjestelmä erottelee puheen taustalla soivasta musiikista. Tallenne on äänitetty hiljaisessa ja kaiuttomassa ympäristössä. Mikrofonijärjestelmä oli asetettu puhujan ja musiikkia toistavien kaiuttimien väliin noin puolen metrin etäisyydelle molemmista.

[2]Valokuva käytetystä mikrofonijärjestelmästä. Kuvaaja: Kalle Palokangas.



Kuva 8. Spektrogrammit mikrofonijärjestelmän tallenteen eri kanavista. Kanavat vasemmalta oikealle: kanava 1: mikrofonin 1 raaka tallenne, kanava 0: mikrofonien 1-4 tallenne, jolle on suoritettu taustamelun vaimennus, kanava 5: toistettu musiikki.

3.3. Ratkaisun kuvaus

Ratkaisussa yhdistetään ReSpeaker Mic Array v2.0 -mikrofonijärjestelmän tuottama prosessoitu kanava (yksikanavaisella laiteohjelmistolla[3]), ja uPIT-syväoppimismenetelmällä koulutettu yksikanavainen puheenerottelumenetelmä, joka on esitelty tarkemmin luvussa 2.3.3.

uPIT-syväoppimismenetelmää käytetään sen algoritmisen yksinkertaisuuden vuoksi verrattuna muihin syväoppimismenetelmiin, kuten syväklusterointi, jonka suoriutuminen on samaa tasoa [70]. Aiemmat tulokset myös osoittavat sen kykenevän ratkaisemaan koulutuksen permutaatio-ongelman tehokkaasti, sekä sillä koulutettujen mallien toimivan pitkälti puhujasta ja kielestä riippumatta [70]. Mallin toimiminen eri kielillä on toteutuksen kannalta oleellista, sillä suomen kielellä ei ole helposti saatavilla yhtä kattavaa kieliaineitoa kuin englannin kielellä. Englanninkielisellä aineistolla koulutetun mallin tulisi siis kyetä erottelemaan myös suomenkielistä puhetta lähes yhtä tehokkaasti.

3.3.1. Ohjelmiston rakenne

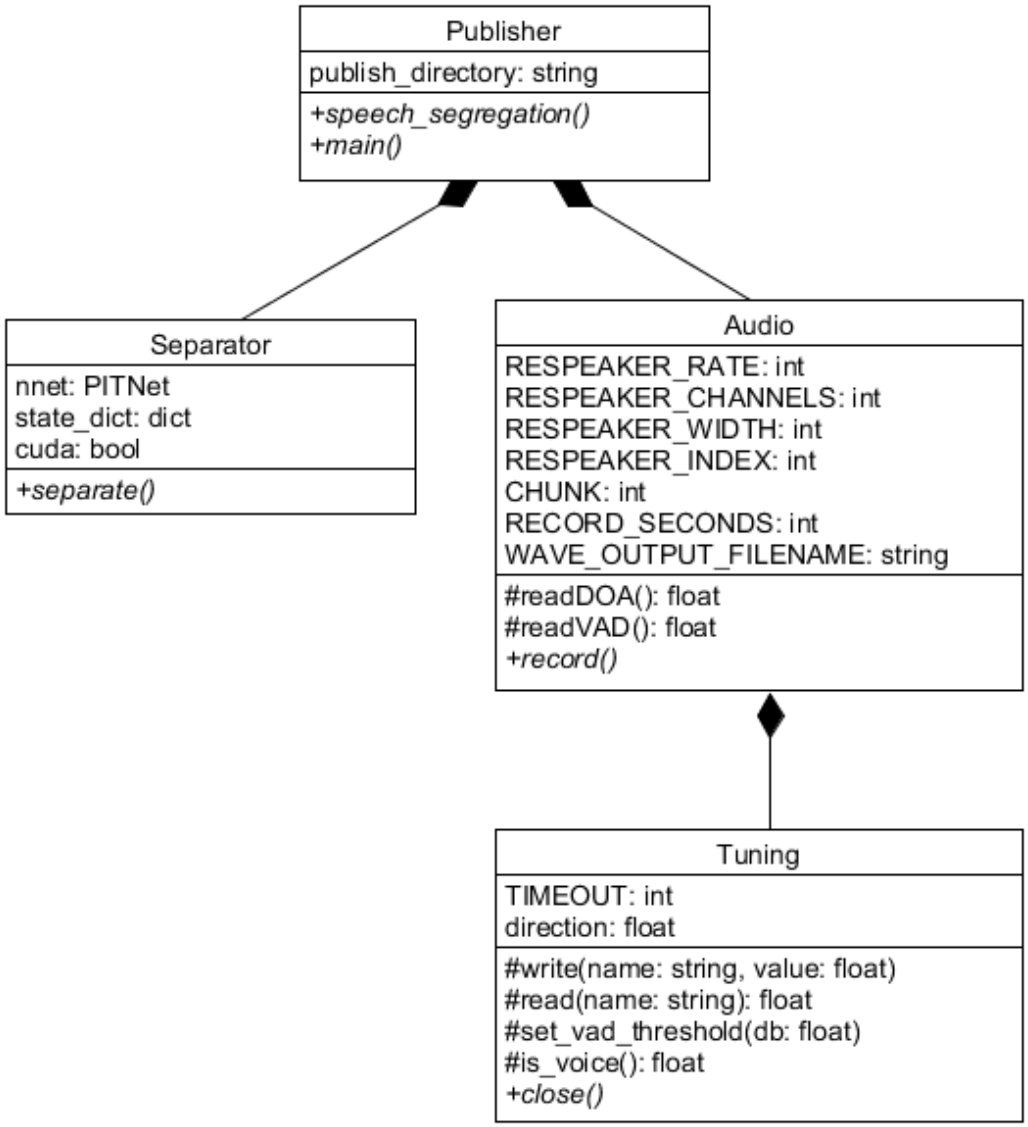
Kuvassa 9 on esitetty toteutetun ohjelmiston rakenne luokkadiagrammilla. ROS-solmu *Publisher* koostuu kahdesta funktiosta, joita suoritetaan rinnakkain eri prosesseissa. Funktio *main* kuuntelee *Audio* luokan avulla mikrofonijärjestelmän DOA ja VAD *(Voice Activity Detection, VAD)* arvoja, ja julkaisee DOA-arvon ROS-aiheeseen, jos VAD on aktiivinen. VAD:n aktivoituessa avataan yhteys mikrofonijärjestelmään, ja tallennetaan ääntä asetuksissa määritellyn tallennuksen keston ajan. Luotu äänite tallennetaan asetuksissa määritettyyn WAV-tiedostoon, joka luetaan toisen prosessin toimesta jatkokäsittelyä varten.

Ohjelman toinen funktio lukee äänitettyjä WAV-tiedostoja, ja uudelleennimeää ne aikaleiman kera, jotta seuraavat äänitteet eivät ylikirjoittaisi niitä. Tieto uusista tiedostoista välitetään *Separator*-luokalle, jossa erottelu suoritetaan taas omassa prosessissaan. Erottelufunktio lataa mallin koulutuksessa käytetyn asetustiedoston,

[3]Mic Array v2.0 laiteohjelmistot: `http://wiki.seeedstudio.com/ReSpeaker_Mic_Array_v2.0/#update-firmware`

sekä koulutuksen tuottaman tilahakemiston, joiden tiedoilla neuroverkkomoduuli *PITNet* alustetaan. Asetustiedosto sisältää tiedot mallin tyypistä, ja tilahakemisto sisältää mallin koulutuksessa opitut parametrit. Erottelun onnistuttua alkuperäinen tiedosto poistetaa, ja erotellut kanavat tallennetaan uusiin WAV-tiedostoihin asetuksissa määritettyyn kansioon esimerkiksi automaattisen puheentunnistuksen käsiteltäväksi.

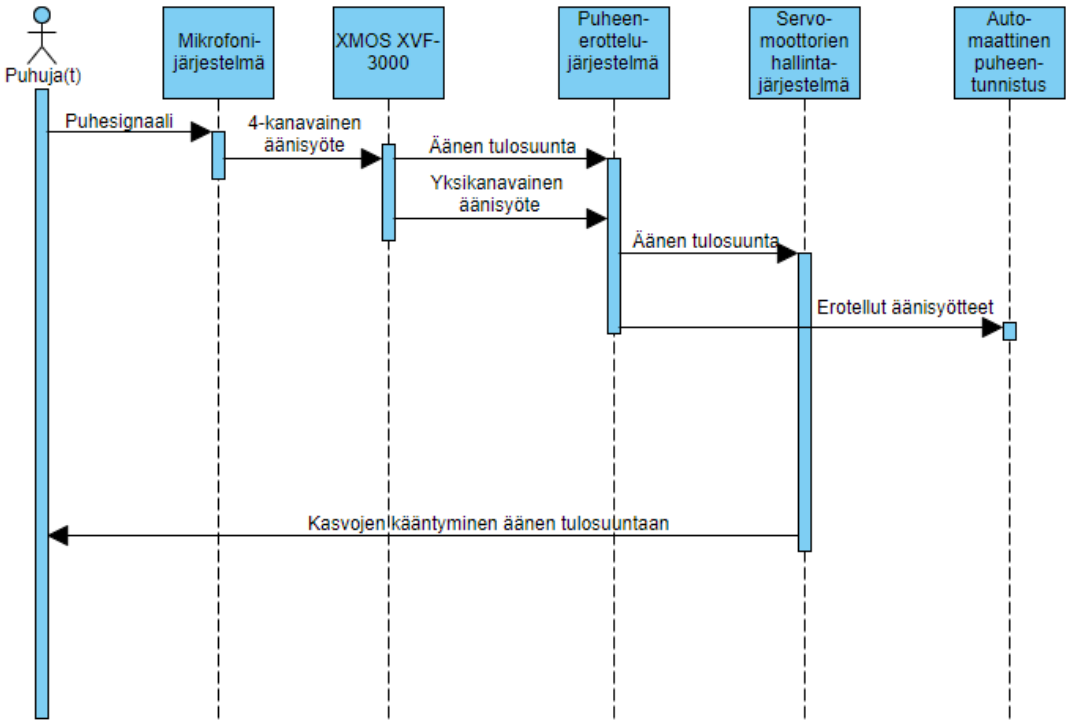
Luokan *Separator* pohja on peräisin Jian Wun toteutuksesta [87], jota käytettiin myös mallin kouluttamiseen. Luokka *Tuning* on peräisin ReSpeaker-laiteohjelmistopaketista [88].



Kuva 9. Ohjelmiston luokkadiagrammi.

Kuvassa 10 on esitetty ylätason kuvaus ohjelman toiminnasta. Puheenerottelujärjestelmä kuvaa toteutettua ohjelmaa, servomoottorien

hallintajärjestelmä ja automaattinen puheentunnistus ovat ulkoisia ROS-solmuja, jotka hyödyntävät ohjelman tulosteita.



Kuva 10. Käyttötapauksen sekvenssikaavio. XMOS XVF-3000 suorittaa sisäänrakennetusti kaiunpoiston, keilanmuodostuksen ja taustamelun vaimennuksen.

3.3.2. Neuroverkon koulutus

Yksikanavaisen puheenerottelun toteutuksessa hyödynnettiin Jian Wun kokeiluja lausahdustason permutaatioinvariantin koulutuksen toteuttamisesta [87] Kolbæk et al. [70] tutkimukseen perustuen.

Kieliaineisto

Wun kokeiluissa, kuten aiemmassa tutkimuksessa [71, 70], käytetään WSJ0-kieliainestosta [89] muodostettua sekoiteaineistoa WSJ0-2mix, jossa kahden puhujan lausahdukset on sekoitettu. Kieliaineisto ei kuitenkaan ole julkisesti saatavissa, vaan vaatii maksullisen lisenssin, joten sama aineisto ei ole käytettävissä tässä työssä. Vaihtoehtona vakiintuneelle WSJ0-aineistolle jouduttiin käyttämään avoimesti saatavilla olevaa LibriSpeech kieliaineistoa [90] vastaavanlaisen sekoiteaineiston muodostamiseen.

Koulutusainesto muodostettiin LibriSpeech ASR -korpuksen sadan tunnin puhtaasta koulutusaineistosta, ja se on hieman laajempi kuin WSJ0-2mix aineiston kolmenkymmenen tunnin koulutusaineisto. Puhujista muodostettiin pareja puhujien puheaineiston keston mukaan niin, että parin molempien puhujien aineistojen

kestot ovat mahdollisimman lähellä toisiaan. Tällä pyritään vähentämään samojen lausahdusten uudelleenkäyttämisen tarvetta puhesekoitteiden muodostamisessa, kun lausahduksia on aineistossa mollemmille puhujille suurin piirtein yhtä paljon. Lausahdusten kestot tosin vaihtelevat paljon, joten osassa sekoitteista on pitkiä osioita, joissa vain toinen puhujista on äänessä. Puhesekoitteiden muodostus toteutettiin Pengin [91] esimerkin pohjalta, ja siitä muodostettiin WSJ0-sekoiteaineiston mukainen metadata, josta sekoitetta vastaavat viitesignaalit, eli alkuperäiset lausahdukset ovat löydettävissä. Koulutusaineistossa on 250 puhujaa, joista 125 on miehiä ja 125 naisia. Koulutuksessa käytettävä sekoiteaineisto sisältää 14819 lausahdusta. Sukupuolet ovat jakautuneet aineistossa niin, että 6702:ssa lausahduksessa puhujat ovat samaa sukupuolta, ja 8117:ssa lausahduksessa vastakkaista sukupuolta.

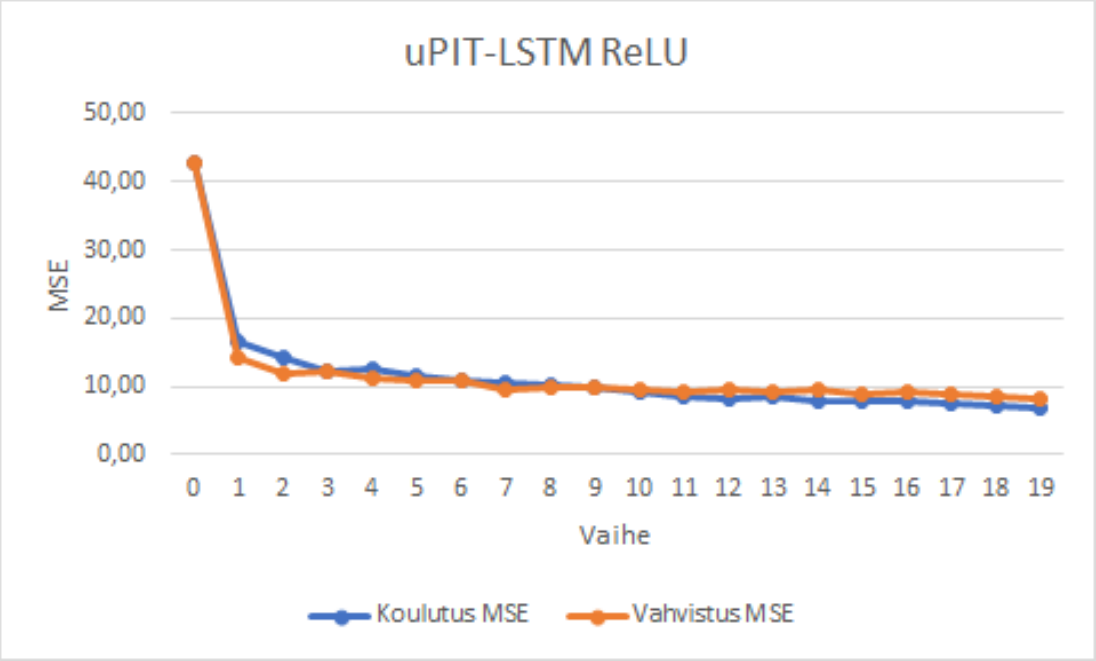
Vahvistus- ja testiaineisto muodosttiin samaan tapaan LibriSpeech ASR -korpuksen puhtaasta kehitysaineistosta, ja puhtaasta testiaineistosta. Vahvistusaineistossa on 40 puhujaa, 20 miestä ja 20 naista, ja se sisältää 1555 lausahdusta. Vahvistusaineistoa käytään koulutettavan mallin alustavien parametrien määrittämiseen, sekä nähdyn puhujan (*Closed Condition, CC*) suorituskyvyn arvioimiseen, samaan tapaan kuin aiemmassa tutkimuksessa [71, 70]. Testiaineistossa on samoin 40 puhujaa joista 20 on miehiä ja 20 naisia, ja se sisältää 1536 lausahdusta. Testiaineiston avulla arvioidaan mallin suorituskykyä uusilla puhujilla (*Open Condition, OC*).

Asetukset

Käytetty malli jäljitteli aiemmassa tutkimuksessa [70] käytettyä yksisuuntaista LSTM RNN -mallia, jossa on kolme 1792 yksikön kokoista kerrosta. Kaksisuuntainen LSTM-malli on osoittautunut aiemmissa kokeiluissa tehokkaammaksi, mutta sen koulutus ei onnistunut käytetyllä laitteistolla näytönohjaimen muistin riittämättömyyden vuoksi, joten toteutuksessa käytetään yksisuuntaista mallia. Koulutuksen nopeuttamiseksi koulutuksessa käytetään Nvidian cuDNN-kirjaston implementaatiota LSTM-mallista. Tästä huolimatta mallin kouluttaminen oli hyvin hidasta, eikä työssä ajanpuutteen vuoksi ehditty kouluttamaan kuin yksi malli. Koulutetussa mallissa käytetään vaiheherkkää maskia *(Phase Sensitive Mask, PSM)*, ja aktivaatiofunktiona käytetään tasasuunnattua lineaariyksikköä *(Rectified Linear Unit, ReLU)*, sillä niitä käyttäen on saatu parhaita tuloksia aiemmissa kokeiluissa [87, 70]. Malli koulutettiin erottelemaan kaksi puhujaa.

Kuvassa 11 on esitetty lausahduskohtaisten keskineliövirheiden keskiarvon kehitys koulutuksen edetessä koulutus- ja vahvistusaineistoilla. Vaiheen nolla keskineliövirhe on alustettu vahvistusaineistosta.

Molempien vahvistus- ja koulutusaineiston MSE-arvojen jatkuva pienentyminen osoittaa uPIT:n ratkaisevan permutaatio-ongelman tehokkaasti. Jokaisessa vaiheessa molemmat aineistot käsitellään uudestaan, joten mallin suorituskykyä voidaan parantaa vaiheiden lisäämisellä koulutuksessa. Vaiheita on koulutettu vain 19 käytetyn laitteiston heikon suorituskyvyn vuoksi, jolla yhden vaiheen kouluttaminen kesti hieman yli neljä tuntia.



Kuva 11. Keskimääräinen lausahduskohtainen MSE koulutuksen eri vaiheissa vahvistus- ja koulutusaineistoissa.

3.4. Sovellusympäristö

Toteutus on suunniteltu sovellusympäristöön, jossa puhuttuja lausahduksia täytyy erotella taustaäänistä, kuten etäisestä puheensorinasta, musiikista, tai muusta melusta, sekä mahdollisesti päällekkäisestä puheesta, eikä puheentunnistuksen reaaliaikaisuuteen pyritä. Kaikki puhe erotellaan puhujasta riippumatta, ja päällekkäisen puheen tapauksessa kaikki puhujat tulee erotella ja välittää erillisinä kanavina jatkokäsittelyyn. Tässä toteutuksessa yhdenaikaisten puhujien määrä on rajattu kahteen mallin yksinkertaistamiseksi.

Sovellusta voidaan käyttää esimerkiksi käyttötapauksessa, jossa InMoov-robotti toimii vuorovaikutteisena sihteerinä, esimerkiksi pienissä, noin 2-10 hengen kokouksissa. Robotin päälaelle kiinnitettävä mikrofonijärjestelmä tunnistaa puheen tulosuunnan ja ohjaa robotin niskan liikkeitä hallinnoivaan ROS-solmua kääntämään robotin kasvot puhujaa kohti. Tämän lisäksi järjestelmä välittää erotellut puhesignaalit automaattiselle puheentunnistukselle, joka suorittaa puheesta tekstiksi muunnoksen, ja tallentaa puheen sisällön pöytäkirjaan.

Sovelluksen osa käyttötapauksessa on mikrofonijärjestelmää hyödyntäen tuottaa ulkoisille ROS-solmuille niiden tarvitsemat syötteet käyttötapauksessa toimimiseen.

3.4.1. Ongelmatilanteet

Vain yhden puhujan ollessa äänessä käyttötapaus on kohtalaisen suoraviivainen. XMOS XVF-3000 -piirisarjaan integroidut digitaalisen signaalikäsittelyn algoritmit suorittavat kaiunpoiston, keilanmuodostuksen, taustamelun suodatuksen ja

vahvistuksensäädön. Tällöin äänen suunta on ainoan puhujan puheen saapumissuunta, ja mikrofonijärjestelmän tuottama käsitelty äänikanava voidaan välittää sellaisenaan automaattiselle puheentunnistukselle. Syväoppimismalli pyrkii kuitenkin aina tunnistamaan puheesta niin monta puhujaa kuin se on koulutettu erottelemaan. Ilman erillistä ratkaisua puhujien määrän tunnistamiseen, sovellus välittää automaattiselle puheentunnistukselle niin monta kanavaa kuin se on koulutettu erottelemaan. Mallin toimiessa oikein, ylimääräisen kanavan tulisi pysyä hiljaisena, kun puhujia on vain yksi [70].

Järjestelmä kohtaa haasteita jos useampi kokouksen osallistuja puhuu yhdenaikaisesti. Jos puhujat ovat eri suunnissa robottiin ja sen mikrofonijärjestelmään nähden, ei äänen suunta ole enään yksiselitteinen. Ennen kuin äänen tulosuunta voidaan välittää robotin päätä ohjaavien servomoottorien hallintajärjestelmään, täytyy määrittää äänen suunta jota kohti käännytään. Robotin päätä voidaan esimerkiksi ohjata kääntymään äänen arvioituun tulosuuntaan välittömästi sen muututtua, tai kun äänen tulosuunta on pysynyt samana tarpeeksi kauan, jotta mahdolliset huudahdukset, tai lyhyet, yhden sanan mittaiset vastaukset kysymyksiin eivät aiheuta pään kääntymistä. Sovellus välittää mikrofonijärjestelmän arvioman äänen tulosuunnan ROS-aiheeseen aina puheen ollessa aktiivinen.

Useat henkilöt voivat puhua robottiin nähden myös suurin piirtein samasta suunnasta, jolloin äänen tulosuunnan määrittämisen sijaan ongelmaksi muodostuu puhujien erotteleminen toisistaan. Käyttötapauksessa puheentunnistuksen ei tarvitse pyrkiä reaaliaikaisuuteen, sillä se tuottaa vain pöytäkirjaa, jota tarvitaan vasta kokouksen päätyttyä. Robotti reagoi vuorovaikutteisesti vain puheen tulosuuntaan, mutta ei sen sisältöön. Näin ollen puhetta voidaan tallentaa esimerkiksi niin kauan kuin sitä havaitaan, jonka jälkeen puheen erottelu suoritetaan tallenteesta, ja välitetään siitä erotellut puheet erillisinä tallenteina automaatiselle puheentunnistukselle puheesta tekstiksi muunnosta varten. Puheen aktiivisuuden havaitseminen sisältyy mikrofonijärjestelmän ominaisuuksiin.

Useiden henkilöiden puhuessa yhdenaikaisesti eri suunnista puheen erottelu ei todennäköisesti onnistu, sillä keilanmuodostus kohdistuu vain yhteen suuntaan, jolloin muusta suunnasta tulevat äänet, puhe mukaanlukien, vaimenevat. Erityisesti jos yhdenaikaisia puhujia on enemmän kuin kaksi, olisi hyvä jos robotti kykenisi ilmaisemaan puhujille ettei se kykene enään ymmärtämään heitä, esimerkiksi pyytämään kokouksen osallistujia vaikenemaan puhesynteesin avulla.

3.5. Mittaustulokset

Erottelualgoritmin suorituskykyä arvioidaan signaali-särösuhteen (*Signal-to-Distortion ratio, SDR*) avulla. SDR on kirjallisuudessa yleisesti käytetty metriikka puheenerottelualgoritmien vertailussa. SDR lasketaan kaavalla

$$SDR = 10 \log_{10} \left(\frac{\|\mathbf{s_{puhdas}}\|_2^2}{\|\hat{\mathbf{s}} - \mathbf{s_{puhdas}}\|_2^2} \right) \text{dB} \tag{6}$$

jossa $\hat{\mathbf{s}}$ on eroteltu puhesignaali ja $\mathbf{s_{puhdas}}$ on verratava alkuperäinen puhesignaalisignaali [15]. Mitä suurempi SDR-arvo sitä paremmin erottelun tuloksena syntynyt signaali vastaa alkuperäistä signaalia.

Taulukossa 1 näkyvät LibriSpeech-aineistolla koulutetun uPIT-LSTM-mallia hyödyntävän erottelualgoritmin tuloksista saadut SDR-arvot. Arvot on laskettu vastakkaista sukupuolta oleville puhujille (NM), miespuolisille puhujille (MM) ja naispuolisille puhujille (NN). Mittaukset on tehty käyttäen vaiheherkkää maskia tasasuunnatulla lineaariyksiköllä (PSM-ReLU) ja vaiheherkkää maskia oraakkeli-informaatiolla (PSM-oracle). Oraakkeli-informaatiolla tehdyt mittaukset käyttävät tietoa alkuperäisistä puhtaista puhesignaaleista ja antavat ne SDR-arvot, jotka saataisiin silloin, kun puhujat on eroteltu ideaalisesti.

Taulukko 1. Erottelumenetelmän SDR-arvot desibeleinä LibriSpeech ASR -korpuksesta muodostetuilla testiaineistoilla sukupuoliryhmittäin. Sulkeissa aineiston sisältämien lausahdusten lukumäärä.

Malli	Maski	CC				OC			
		NM (696)	MM (422)	NN (437)	YHT (1555)	NM (814)	MM (333)	NN (389)	YHT (1536)
uPIT-LSTM	PSM-ReLU	5.99	3.80	3.19	4.32	5.60	3.40	2.18	3.72
-	PSM-oracle	15.93	14.86	16.50	15.76	15.46	14.75	15.81	15.34

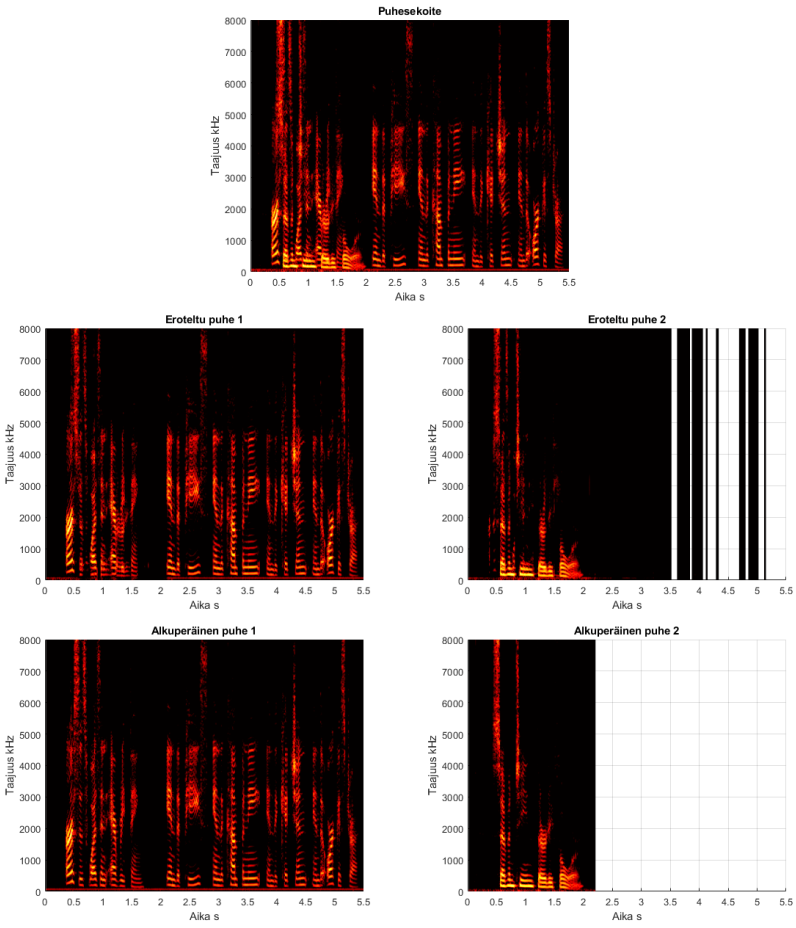
Mittauksista nähdään, että molemmilla, sekä jo nähdyillä puhujilla (CC) että uusilla puhujilla (OC), vastakkaisten sukupuolten SDR-arvot ovat suurempia kuin vastaavilla samaa sukupuolta olevilla puhujilla. Tulos oli odotettavissa, sillä esimerkiksi Qian et al. [13] mukaan syväoppimiseen perustuvat erottelumenetelmät suoriutuvat huomattavasti paremmin, kun puhujat ovat vastakkaista sukupuolta. Koulutusdata oli myös jakautunut epätasaisesti niin, että NM-pareja oli 8117 kpl, MM-pareja 3291 kpl ja NN-pareja 3411 kpl. Malli siis altistui useammin vastakkaista sukupuolta oleville puhujille kuin samaa sukupuolta oleville puhujille, joka mahdollisesti myös selittää eroa puhujaparien SDR-arvojen välillä.

Kun mittaustuloksia verrataan muiden kirjallisuudessa esitettyjen mallien suorituskykyyn tulokset vaikuttavat lupaavilta: Esimerkiksi Qian et al. [13] esittämissä tuloksissa perinteinen yksikanavainen CASA-menetelmä saavutti jo nähdyille puhujille 3,1 dB ja uusille puhujille 2,9 dB parannuksen signaali-särösuhteessa. Kuvan permutaatioinvariantti koulutus tuotti puolestaan sekä jo nähdyille että uusille puhujille 10,0 dB parannuksen. Huomioon otettavaa on kuitenkin se, että Qianin artikkelissa esitetyt menetelmät on koulutettu käyttäen WSJ0-2mix-aineistoa LibriSpeech-aineiston sijaan. Arvot eivät siis ole täysin vertailukelpoisia. Suuntaa antavasti voidaan kuitenkin todeta, että kehitetyn mallin suorituskyky ei vielä yllä muiden PIT-toteutusten tasolle, mutta näyttäisi antavan parempia tuloksia ainakin CASA:an verrattuna. Saadut SDR-arvot jäävät myös kohtalaisen kauas oraakkeli-informaatiolla lasketuista ideaaliarvoista, joten lisää kehitystä tarvitaan.

4. POHDINTA

uPIT:llä koulutettu LSTM-malli tuotti lupaavia tuloksia verrattaen vähäiseksi jääneeseen koulutukseen nähden. Tulokset eivät kuitenkaan ole niin merkittäviä, että syväoppimismenetelmän käyttäminen olisi perusteltua ympäristössä, jossa laskentatehoa ja muistia on rajoitetusti. Toteutetun ohjelmiston kirjastoriippuvuuksista erityisesti PyTorch ja LibROSA ovat niin suuria, että niiden sisällyttäminen sulautetuille laitteille, kuten Raspberry Pi:lle ei ole suositeltavaa.

Neuroverkon koulutuksessa käytettävää aineistoa tulisi myös laajentaa. Luodun puhesekoiteaineiston sisältämien lausahdusten kesto vaikuttaa vaihtelevan paljon, jolloin aineistoon syntyy paljon sekoitteita, joissa päällekkäisen puheen osuus on lyhyt. Kuvassa 12 esitetyt spektrogrammit havainnollistavat, kuinka erimittaisista lausahduksista luotu puhesekoite on eroteltu koulutetulla mallilla. Kuvassa näkyy, kuinka koulutettu malli kykenee tunnistamaan puhujat ja suodattamaan puhujan yksi äänen pois puhujan kaksi tulosteesta, kun puhuja kaksi ei ole enään äänessä. Päällekkäisten osuuksien erottelussa malli ei kuitenkaan vielä suoriudu yhtä vakuuttavasti. Havainnollistuksessa käytetty puhesekoite on osa vahvistusainestoa, ja siitä eroteltujen tulosteiden SDR-keskiarvo on 13,08 desibeliä.



Kuva 12. Spektrogrammit, joista näkyy kuinka uPIT:llä koulutettu LSTM-malli erottelee kahden puhujan sekoitteen.

4.1. Jatkokehitys

Toteutetun ohjelmiston kirjastoriippuvuuksista erityisesti PyTorch ja LibROSA ovat niin laajoja ettei toteutetun ohjelmiston suorittaminen sulautetuilla laitteilla, kuten Raspberry Pi:llä, ole käytännöllistä. Ohjelmasta tulisi tehdä erillinen, riisuttu versio sulautetuille laitteille ilman yksikanavaista puheenerottelua ja sen vaatimia kirjastoriippuvuuksia. Tällöin ohjelmaa voisi yhä käyttää tapauksissa, joissa päällekkäisen puheen erottelu ei ole välttämätöntä. Kirjastoriippuvuuksia voisi myös pyrkiä korvaamaan kevyemmillä ratkaisuilla.

Yksikanavaisen puheenerottelun suorituskykyä voidaan parantaa lisäämällä koulutuksen vaiheita. uPIT käsittelee jokaisen otoksen vain kerran jokaista vaihetta kohden, joten koulutusta on aiemmin jatkettu jopa 200 vaiheen verran [70]. Käytetyn yksisuuntaisen LSTM-mallin sijaan voitaisiin myös käyttää kaksisuuntaista LSTM-mallia, jonka tulisi suoriutua paremmin [70]. Malli voitaisiin myös kouluttaa kolmelle puhujalle, sekä lisätä koulutusaineistoon myös suomenkielisiä puhesekoitteita, joka mahdollisesti parantaisi mallin suorituskykyä suomenkielisen puheen erottelemisessa.

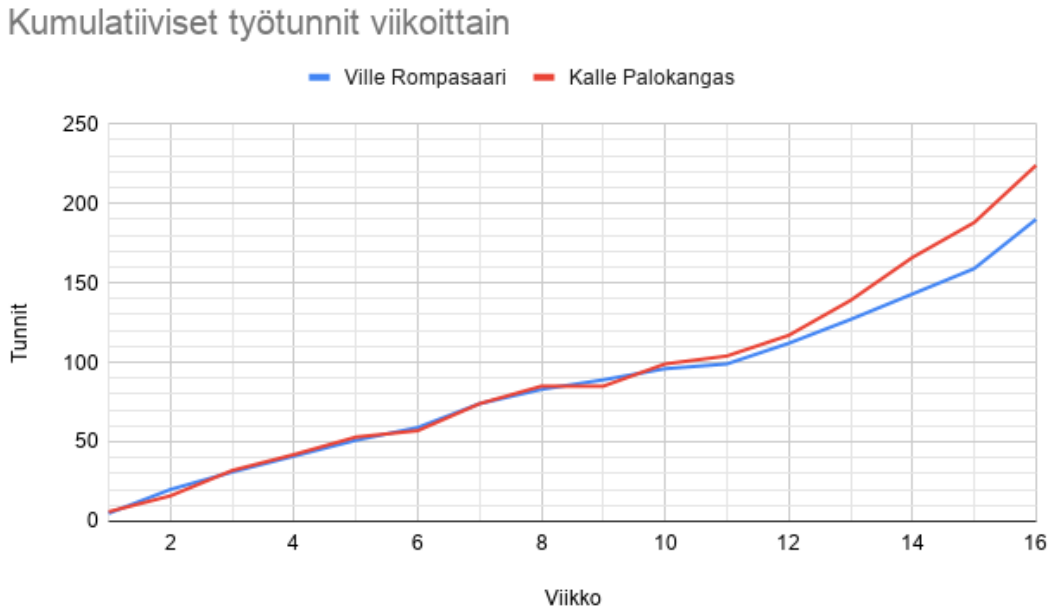
Merkittävä parannus ohjelman suorituskykyyn olisi tunnistaa tilanteet, joissa vain yksi puhuja on äänessä. Tällöin yksikanavainen puheenerottelu voitaisiin ohittaa, ja välittää vain mikrofonijärjestelmän käsittelemä audio suoraan automaattiselle puheentunnistukselle. Informaatiota puhujien määrästä voitaisiin saada esimerkiksi robotin konenäön välityksellä, joka voisi tunnistaa robottia kohti puhuvien kasvojen lukumäärän, ja välittää sen puheenerottelujärjestelmälle. Ilman tietoa puhujien määrästä puheenerottelu joudutaan suorittamaan jokaiselle tallenteelle, vaikka se ei monissa tilanteissa olisikaan tarpeellista.

Robotin konenäön antamia arvioita puhujien suunnista voitaisiin myös yhdistää mikrofonijärjestelmän laskemiin suuntaestimaatteihin, joka parantaisi konekuulon kykyä reagoida muuttuviin tilanteisiin. Tarkempia ja nopeampia suuntaestimaatteja voitaisiin hyödyntää esimerkiksi mikrofonijärjestelmän keilanmuodostuksen suuntaamisessa. Lisäksi mikrofonijärjestelmän XMOS XVF-3000 -piirisarjan toiminta kannattaisi räätälöidä käyttötapaukseen sopivaksi tai korvata se kokonaan erillisellä toteutuksella, joka käsittelisi yksittäisten mikrofonien tuottamia signaaleja. Käyttövalmiina mikrofonijärjestelmä kykenee vain yhden puhujan suunnan arviointiin mutta erillisellä toteutuksella sitä voitaisiin laajentaa mahdollisesti myös useiden puhujien suuntien seuraamiseen.

5. PROJEKTIN KUVAUS

Työskentely tapahtui osin ryhmässä ja osin itsenäisesti. Ryhmä tapasi useita kertoja viikossa Oulun yliopiston tiloissa tai etäyhteyden välityksellä sekä kurssin ohjaustapaamisissa että itsenäisesti. Tapaamisissa keskusteltiin projektin kulusta ja sisällöstä, määritettiin työn seuraavat vaiheet ja suunniteltiin niiden työnjako. Ryhmän molemmat jäsenet työskentelivät kaikilla työn osa-alueilla.

Projektin aikana ylläpidettiin viikoittaista ajankäytön seurantaa, joka on esitetty kuvassa 13. Projektiin käytetty työmäärä jakaantui melko tasaisesti koko projektin ajalle ja sitä kertyi henkilöä kohden noin 200 tuntia.



Kuva 13. Projektin ajankäytön seuranta.

6. YHTEENVETO

Tässä kandidaatintyössä esiteltiin uPIT-syväoppimismenetelmää hyödyntävä toteutus puheenerottelujärjestelmästä. Järjestelmä kehitettiin ROS-käyttöjärjestelmää käyttävälle InMoov-humanoidirobotille Sulautettujen ohjelmistojen projekti -kurssin yhteydessä. Työn tavoitteena oli pyrkiä luomaan ne osat robotin konekuulojärjestelmästä, jotka toteuttavat äänen tallennuksen, melun- ja kaiunpoiston, äänilähteen suunnan määrityksen ja puheenerottelun. Robotille valittu neljästä mikrofonista koostuva mikrofonijärjestelmä tarjoaa järjestelmälle käyttövalmiina melun- ja kaiunpoiston, keilanmuodostuksen ja voimakkaimman äänilähteen tulosuunnan. Mikrofonin tallentama käsitelty äänisignaali ja suunnan estimaatti välitetään kehitetylle järjestelmälle, jossa äänisignaalin sisältämät puhesignaalit pyritään erottamaan toisistaan ja välittämään suuntaestimaatin ohella muiden robotin ROS-solmujen saataville. Mikrofonijärjestelmää täydennetään LibriSpeech-kieliaineistolla koulutetulla uPIT-LSTM-mallia hyödyntävällä yksikanavaisella puheenerottelualgoritmilla.

Toteutus mahdollistaa meluisassa tai kaikuvassa ympäristössä toimivan puheentunnistusmenetelmän toteuttamisen InMoov-robotille. Koulutettu syväoppimismalli ei vielä nykyisellään sovellu käytettäväksi sulautetuilla laitteilla, mutta se tarjoaa hyvän lähtökohdan puhujien erottelemiseen.

7. VIITTEET

[1] Barker J., Marxer R., Vincent E. & Watanabe S. (2017) The third ‘chime’ speech separation and recognition challenge: Analysis and outcomes. Computer Speech & Language 46, ss. 605 – 626.

[2] Attawibulkul S., Kaewkamnerdpong B. & Miyanaga Y. (2017) Noisy speech training in mfcc-based speech recognition with noise suppression toward robot assisted autism therapy. nide 2017-January, ss. 1–5.

[3] Ruzaij M.F., Neubert S., Stoll N. & Thurow K. (2017) Design and implementation of low-cost intelligent wheelchair controller for quadriplegias and paralysis patient. Teoksessa: 2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI), ss. 000399–000404.

[4] Merks I., Enzner G. & Zhang T. (2013) Sound source localization with binaural hearing aids using adaptive blind channel identification. Teoksessa: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, ss. 438–442.

[5] Archer-Boyd A., Whitmer W., Brimijoin W. & Soraghan J. (2015) Biomimetic direction of arrival estimation for resolving front-back confusions in hearing aids. Journal of the Acoustical Society of America 137, ss. EL360–EL366.

[6] Loh C., Boey K. & Hong K. (2017) Speech recognition interactive system for vehicle. ss. 85–88.

[7] Dahlan B., Mansoor W., Abbasi M. & Honarbakhsh P. (2011) Sound source localization for automatic camera steering. Teoksessa: The 7th International Conference on Networked Computing and Advanced Information Management, ss. 20–25.

[8] Chung H., Iorga M., Voas J. & Lee S. (2017) Alexa, can i trust you? Computer 50, ss. 100–104.

[9] Lyon R. (2010) Machine hearing: An emerging field. IEEE Signal Processing Magazine 27, ss. 131–135+139.

[10] Cherry E. (1953) Some experiments on the recognition of speech, with one and with two ears. Journal of the Acoustical Society of America 25, ss. 975–979.

[11] Takiguchi T., Sumida Y., Takashima R. & Ariki Y. (2009) Single-channel talker localization based on discrimination of acoustic transfer functions. Eurasip Journal on Advances in Signal Processing 2009.

[12] Fuchs A., Feldbauer C. & Stark M. (2011) Monaural sound localization. ss. 2521–2524.

[13] Qian Y.M., Weng C., Chang X.K., Wang S. & Yu D. (2018) Past review, current progress, and challenges ahead on the cocktail party problem. Frontiers of Information Technology and Electronic Engineering 19, ss. 40–63.

[14] Kendall G.S. (1995) 3-d sound primer: directional hearing and stereo reproduction. Computer Music Journal 19, ss. 23–46.

[15] Mahkonen K. (2018) Efficient and robust methods for audio and video signal analysis. väitöskirja, Tampereen teknillinen yliopisto.

[16] Naylor P.A. & Gaubitch N.D. (2010) Introduction, Springer London, London. ss. 1–19.

[17] Kinoshita K., Delcroix M., Yoshioka T., Nakatani T., Sehr A., Kellermann W. & Maas R. (2013) The reverb challenge: Acommon evaluation framework for dereverberation and recognition of reverberant speech.

[18] Ideli E., Vaughan R. & Bajic I. (2018) Speech intelligibility of microphone arrays in reverberant environments with interference.

[19] Wang D. & Chen J. (2018) Supervised speech separation based on deep learning: An overview. IEEE/ACM Transactions on Audio Speech and Language Processing 26, ss. 1702–1726.

[20] Han K., Wang Y. & Wang D. (2014) Learning spectral mapping for speech dereverberation. Teoksessa: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), ss. 4628–4632.

[21] Pados D.A. & Karystinos G.N. (2001) An iterative algorithm for the computation of the mvdr filter. IEEE Transactions On signal processing 49, ss. 290–300.

[22] Wan X. & Wu Z. (2013) Sound source localization based on discrimination of cross-correlation functions. Applied Acoustics 74, ss. 28 – 37.

[23] Sahley T.L. & Musiek F.E. (2015) Basic Fundamentals in Hearing Science. Plural Publishing, Inc.

[24] Chen J., Benesty J. & Huang Y. (2006) Time delay estimation in room acoustic environments: An overview. Eurasip Journal on Applied Signal Processing 2006.

[25] Brandstein M.S. & Silverman H.F. (1997) A practical methodology for speech source localization with microphone arrays. Computer Speech & Language 11, ss. 91 – 126.

[26] Schmidt R. (1986) Multiple emitter location and signal parameter estimation. IEEE Transactions on Antennas and Propagation 34, ss. 276–280.

[27] He J., Swamy M.N.S. & Ahmad M.O. (2012) Efficient application of music algorithm under the coexistence of far-field and near-field sources. IEEE Transactions on Signal Processing 60, ss. 2066–2070.

[28] Zhao S., Saluev T. & Jones D.L. (2014) Underdetermined direction of arrival estimation using acoustic vector sensor. Signal Processing 100, ss. 160 – 168.

[29] Ebrahimi A.A., Abutalebi H.R. & Karimi M. (2019) Generalised two stage cumulants-based music algorithm for passive mixed sources localisation. IET Signal Processing 13, ss. 409–414.

[30] Rascon C., Meza I., Fuentes G., Salinas L. & Pineda L. (2015) Integration of the multi-doa estimation functionality to human-robot interaction. International Journal of Advanced Robotic Systems 12.

[31] Dmochowski J.P., Benesty J. & Affes S. (2007) A generalized steered response power method for computationally viable source localization. IEEE Transactions on Audio, Speech, and Language Processing 15, ss. 2510–2526.

[32] Diaz-Guerra D. & Beltran J.R. (2018) Direction of arrival estimation with microphone arrays using srp-phat and neural networks. Teoksessa: 2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM), ss. 617–621.

[33] Blandin C., Ozerov A. & Vincent E. (2012) Multi-source tdoa estimation in reverberant audio using angular spectra and clustering. Signal Processing 92, ss. 1950 – 1960. Latent Variable Analysis and Signal Separation.

[34] Brutti A., Omologo M. & Svaizer P. (2008) Comparison between different sound source localization techniques based on a real data collection. ss. 69–72.

[35] Khaykin D. & Rafaely B. (2009) Coherent signals direction-of-arrival estimation using a spherical microphone array: Frequency smoothing approach. ss. 221–224.

[36] Xiao X., Zhao S., Zhong X., Jones D.L., Chng E.S. & Li H. (2015) A learning-based approach to direction of arrival estimation in noisy and reverberant environments. Teoksessa: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), ss. 2814–2818.

[37] Vesperini F., Vecchiotti P., Principi E., Squartini S. & Piazza F. (2018) Localizing speakers in multiple rooms by using deep neural networks. Computer Speech and Language 49, ss. 83–106.

[38] Wang Z., Zhang X. & Wang D. (2019) Robust speaker localization guided by deep learning-based time-frequency masking. IEEE/ACM Transactions on Audio, Speech, and Language Processing 27, ss. 178–188.

[39] Ma N., May T. & Brown G.J. (2017) Exploiting deep neural networks and head movements for robust binaural localization of multiple sources in reverberant environments. IEEE/ACM Transactions on Audio, Speech, and Language Processing 25, ss. 2444–2453.

[40] Møller A.R. (2013) Hearing : Anatomy, Physiology, and Disorders of the Auditory System., nide Third edition. Plural Publishing, Inc.

[41] Ma N., May T., Wierstorf H. & Brown G.J. (2015) A machine-hearing system exploiting head movements for binaural sound localisation in reverberant conditions. Teoksessa: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), ss. 2699–2703.

[42] Hornstein J., Lopes M., Santos-Victor J. & Lacerda F. (2006) Sound localization for humanoid robots - building audio-motor maps based on the hrtf. Teoksessa: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, ss. 1170–1176.

[43] Haykin S. & Chen Z. (2005) The cocktail party problem. Neural Computation 17, ss. 1875–1902.

[44] Qazi K., Nawaz T., Mehmood Z., Rashid M. & Habib H. (2018) A hybrid technique for speech segregation and classification using a sophisticated deep neural network. PLoS ONE 13.

[45] Wang Y., Han K. & Wang D. (2013) Exploring monaural features for classification-based speech segregation. IEEE Transactions on Audio, Speech and Language Processing 21, ss. 270–279.

[46] Hu G. & Wang D. (2004) Monaural speech segregation based on pitch tracking and amplitude modulation. IEEE Transactions on neural networks 15, ss. 1135–1150.

[47] Han K. & Wang D. (2012) A classification based approach to speech segregation. Journal of the Acoustical Society of America 132, ss. 3475–3483.

[48] Shao Y., Srinivasan S., Jin Z. & Wang D. (2010) A computational auditory scene analysis system for speech segregation and robust speech recognition. Computer Speech and Language 24, ss. 77–93.

[49] Zhao X., Shao Y. & Wang D. (2012) Casa-based robust speaker identification. IEEE Transactions on Audio, Speech and Language Processing 20, ss. 1608–1616.

[50] Wiem B., Anouar B.M.M. & Aicha B. (2016) Soft-casa system for single channel speech separation. Teoksessa: 2016 4th International Conference on Control Engineering Information Technology (CEIT), ss. 1–5.

[51] Hu G. & Wang D. (2008) Segregation of unvoiced speech from nonspeech interference. The Journal of the Acoustical Society of America 124, ss. 1306–1319.

[52] Lee D.D. & Seung H.S. (2001) Algorithms for non-negative matrix factorization. Teoksessa: Advances in neural information processing systems, ss. 556–562.

[53] Mohammadiha N., Smaragdis P. & Leijon A. (2013) Supervised and unsupervised speech enhancement using nonnegative matrix factorization. IEEE Transactions on Audio, Speech and Language Processing 21, ss. 2140–2151.

[54] Schmidt M.N. & Olsson R.K. (2006) Single-channel speech separation using sparse non-negative matrix factorization. Teoksessa: Ninth International Conference on Spoken Language Processing.

[55] Virtanen T. (2007) Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. IEEE transactions on audio, speech, and language processing 15, ss. 1066–1074.

[56] Li Y., Zhang X. & Sun M. (2017) Robust non-negative matrix factorization with β-divergence for speech separation. ETRI Journal 39, ss. 21–29.

[57] Févotte C. & Idier J. (2011) Algorithms for nonnegative matrix factorization with the β-divergence. Neural computation 23.

[58] Chen Z., McFee B. & Ellis D.P. (2014) Speech enhancement by low-rank and convolutive dictionary spectrogram decomposition. Teoksessa: Fifteenth Annual Conference of the International Speech Communication Association.

[59] Chen Z., Yoshioka T., Xiao X., Li L., Seltzer M. & Gong Y. (2018) Efficient integration of fixed beamformers and speech separation networks for multi-channel far-field speech separation. nide 2018-April, ss. 5384–5388.

[60] Wang Z.Q., Le Roux J. & Hershey J. (2018) Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation. nide 2018-April, ss. 1–5.

[61] Cardoso J.F. (1998) Blind signal separation: Statistical principles. Proceedings of the IEEE 86, ss. 2009–2025.

[62] Jutten C. & Comon P. (2010) Chapter 1 - introduction. Teoksessa: P. Comon & C. Jutten (toim.) Handbook of Blind Source Separation, Academic Press, Oxford, ss. 1 – 22.

[63] Vincent E., Gribonval R. & Févotte C. (2006) Performance measurement in blind audio source separation. IEEE Transactions on Audio, Speech and Language Processing 14, ss. 1462–1469.

[64] Tharwat A. (2018) Independent component analysis: An introduction. Applied Computing and Informatics .

[65] Cao X.R. & Liu R.W. (1996) General approach to blind source separation. IEEE Transactions on Signal Processing 44, ss. 562–571.

[66] Hyvärinen A. & Oja E. (2000) Independent component analysis: Algorithms and applications. Neural Networks 13, ss. 411–430.

[67] Shlens J. (2014) A tutorial on independent component analysis. arXiv preprint arXiv:1404.2986 .

[68] Chen Z., Luo Y. & Mesgarani N. (2017) Deep attractor network for single-microphone speaker separation. ss. 246–250.

[69] Chen Z., Li J., Xiao X., Yoshioka T., Wang H., Wang Z. & Gong Y. (2018) Cracking the cocktail party problem by multi-beam deep attractor network. nide 2018-January, ss. 437–444.

[70] Yu D., Kolbæk M., Tan Z. & Jensen J. (2017) Permutation invariant training of deep models for speaker-independent multi-talker speech separation. Teoksessa: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), ss. 241–245.

[71] Kolbæk M., Yu D., Tan Z.H. & Jensen J. (2017) Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks. IEEE/ACM Transactions on Audio Speech and Language Processing 25, ss. 1901–1913.

[72] Hershey J., Chen Z., Le Roux J. & Watanabe S. (2016) Deep clustering: Discriminative embeddings for segmentation and separation. nide 2016-May, ss. 31–35.

[73] Wang Z.Q., Roux J. & Hershey J. (2018) Alternative objective functions for deep clustering. nide 2018-April, ss. 686–690.

[74] O'Shaughnessy D. (2008) Invited paper: Automatic speech recognition: History, methods and challenges. PATTERN RECOGNITION 41, ss. 2965–2979.

[75] Benzeghiba M., De Mori R., Deroo O., Dupont S., Erbes T., Jouvet D., Fissore L., Laface P., Mertins A., Ris C., Rose R., Tyagi V. & Wellekens C. (2007) Automatic speech recognition and speech variability: A review. Speech Communication 49, ss. 763–786.

[76] RABINER L. (1989) A TUTORIAL ON HIDDEN MARKOV-MODELS AND SELECTED APPLICATIONS IN SPEECH RECOGNITION. PROCEEDINGS OF THE IEEE 77, ss. 257–286.

[77] Vojtas P., Stepan J., Sec D., Cimler R. & Krejcar O. (2018) Voice recognition software on embedded devices. Teoksessa: N.T. Nguyen, D.H. Hoang, T.P. Hong, H. Pham & B. Trawiński (toim.) Intelligent Information and Database Systems, Springer International Publishing, Cham, ss. 642–650.

[78] Molitch-Hou M. (2013), Inmoov project. http://inmoov.fr/project/. Haettu: 12.4.2020.

[79] Fernandez E. (2015) Learning ROS for Robotics Programming - Second Edition., Community Experience Distilled, nide Second edition. Packt Publishing.

[80] Paszke A., Gross S., Massa F., Lerer A., Bradbury J., Chanan G., Killeen T., Lin Z., Gimelshein N., Antiga L., Desmaison A., Kopf A., Yang E., DeVito Z., Raison M., Tejani A., Chilamkurthy S., Steiner B., Fang L., Bai J. & Chintala S. (2019) Pytorch: An imperative style, high-performance deep learning library. Teoksessa: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox & R. Garnett (toim.) Advances in Neural Information Processing Systems 32, Curran Associates, Inc., ss. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[81] McFee B., Lostanlen V., McVicar M., Metsai A., Balke S., Thomé C., Raffel C., Malek A., Lee D., Zalkow F., Lee K., Nieto O., Mason J., Ellis D., Yamamoto R., Seyfarth S., Battenberg E., Виктор Морозов, Bittner R., Choi K., Moore J., Wei Z., Hidaka S., nullmightybofo, Friesch P., Stöter F.R., Hereñú D., Kim T., Vollrath M. & Weiss A. (2020), librosa/librosa: 0.7.2. URL: https://doi.org/10.5281/zenodo.3606573.

[82] Virtanen P., Gommers R., Oliphant T.E., Haberland M., Reddy T., Cournapeau D., Burovski E., Peterson P., Weckesser W., Bright J., van der Walt S.J., Brett M., Wilson J., Jarrod Millman K., Mayorov N., Nelson A.R.J., Jones E., Kern R., Larson E., Carey C., Polat İ., Feng Y., Moore E.W., Vand erPlas J., Laxalde D., Perktold J., Cimrman R., Henriksen I., Quintero E.A., Harris C.R., Archibald A.M., Ribeiro A.H., Pedregosa F., van Mulbregt P. & Contributors S... (2020) SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods 17, ss. 261–272.

[83] Conley K., rospy. http://wiki.ros.org/rospy. Haettu: 22.4.2020.

[84] Pham H., Pyaudio. http://people.csail.mit.edu/hubert/pyaudio/. Haettu: 23.4.2020.

[85] Benesty Jacob k. (2015) Design of circular differential microphone arrays. Springer topics in signal processing, Springer, Cham Switzerland.

[86] Furuta Y., respeaker_ros. https://github.com/furushchev/respeaker_ros. Haettu: 28.4.2020.

[87] Wu J. (2018), Speech separation with utterance-level pit experiments. https://github.com/funcwj/uPIT-for-speech-separation. Haettu: 14.4.2020.

[88] jerryyip, Respeaker 4 mic array. https://github.com/respeaker/usb_4_mic_array. Haettu: 23.4.2020.

[89] Consortium P.L.D., CSR-I (WSJ0) Complete. https://catalog.ldc.upenn.edu/LDC93S6A. Haettu: 15.4.2020.

[90] Panayotov V., Chen G., Povey D. & Khudanpur S. (2015) Librispeech: An asr corpus based on public domain audio books. Teoksessa: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), ss. 5206–5210.

[91] Chao Peng EECS P.U. (2018), Two-talker speech separation with lstm/blstm by permutation invariant training method. https://github.com/pchao6/LSTM_PIT_Speech_Separation. Haettu: 16.4.2020.